



**UNIVERSIDAD NACIONAL AUTÓNOMA DE
MÉXICO**

FACULTAD DE INGENIERÍA

**MANIPULADOR MÓVIL: ESTUDIO SOBRE LA
COORDINACIÓN DE MOVIMIENTOS DE UN
MANIPULADOR SERIAL ACOPLADO**

T E S I S
Que para obtener el título de
INGENIERO MECATRÓNICO

P R E S E N T A
PEDRO MARIANO ESCOBEDO CASTILLO

DIRECTOR DE TESIS:
DR. VÍCTOR JAVIER GONZÁLEZ VILLELA

Agosto 2012





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.

DEDICATORIAS

A mi mamá Ana Lilia por siempre estar ahí para mí,
a mi papá Pedro por su apoyo imprescindible.
Gracias a ustedes he llegado hasta aquí.

A mis hermanos Carlos y Daniel,
por todo lo que hemos vivido.

A mi novia Angélica por haberme brindado su amor
y comprensión a lo largo de la carrera.
A su familia por todo su apoyo.

A mi abuela, tí@s y prim@s por su apoyo
y seguimiento de mis pasos con afecto.

A mis amigos por su apoyo incondicional,
experiencias vividas y solidaridad.

AGRADECIMIENTOS

Agradezco a la Universidad Nacional Autónoma de México por la excelente formación profesional que me ha brindado.

A los profesores de la Facultad de Ingeniería que me brindaron el conocimiento necesario para lograr mi formación profesional.

Al Dr. Víctor Javier González Villela, quien no solo me dio la oportunidad de realizar una tesis, sino que me tuvo la paciencia y el apoyo para concluirla exitosamente.

Agradezco en lo que le corresponde a la DGAPA, UNAM, por el apoyo brindado para la realización de este trabajo, a través de los proyectos PAPIIT IN108308:

“Investigación y desarrollo en robótica móvil, robótica paralela y sistemas mecatrónicos (2008-2010)” y PAPIIT IN115811: “Investigación y desarrollo en sistemas mecatrónicos: robótica móvil, robótica paralela, robótica híbrida y teleoperación (2011-2013)”.

RESUMEN

En el presente trabajo se muestra el estudio sobre la coordinación de movimientos de un manipulador móvil. A su vez se selecciona el tipo de manipulador móvil a emplear y se propone la tarea de recoger un objeto y colocarlo en otra posición. Seguidamente se realiza el diseño de un prototipo de manipulador móvil en el *software* de CAD *Unigraphics*, que consiste de un manipulador serial acoplado a un robot móvil. Se realiza el análisis cinemático tanto del manipulador como del robot móvil, seguido de las ecuaciones que acoplan ambas partes. Se describe un algoritmo heurístico propuesto para la coordinación de movimientos de ambas partes del manipulador móvil, se realiza la planeación de movimientos mediante la operación *pick and place* para el manipulador y por campos potenciales para el robot móvil. Los datos de posición se adquieren mediante *reactIVision*, éste a su vez los envía a *Simulink* para ser procesados; siendo *Simulink* el software donde se implementa la coordinación de movimientos. *Simulink* envía los datos para las juntas del manipulador y velocidad de llantas por puerto serial al micro-controlador Arduino BT y de aquí, mediante el protocolo I^2C , se mandan los comandos a la tarjeta MD25 encargada del movimiento de las llantas.

Finalmente se presentan los resultados de la prueba realizada para comprobar la coordinación de movimientos. En los resultados se muestra el funcionamiento de cada parte, pues ambas trabajan de forma coordinada para cumplir con la tarea especificada: demostrando que el algoritmo propuesto es funcional y se puede aplicar para realizar otro tipo de tareas e incluso implementar a otro tipo de manipuladores móviles.

CONTENIDO

CONTENIDO.....	I
LISTA DE TABLAS Y FIGURAS	IV
I. INTRODUCCIÓN	1
II. MANIPULACIÓN MÓVIL	7
II. 1. Tipos de Manipuladores Móviles	7
II. 1. 1. Manipulador Móvil de Manipulador Humanoide.	8
II. 1. 2. Manipulador Móvil de Brazo Manipulador	9
II. 2. Enfoques sobre coordinación de movimientos	11
II. 2. 1. Enfoque de redes neuronales [1].....	11
II. 2. 2 Enfoque cinemático del jacobiano [2].....	13
II. 2. 3. Enfoque del vehículo virtual [4].....	14
II. 2. 4. Enfoque de la configuración preferida del manipulador [6].....	15
II. 3. Manipulador Móvil seleccionado.	16
II. 4. Diseño de la tarea a desempeñar	17
III. DISEÑO DEL MANIPULADOR MÓVIL.....	19
III. 1. Diseño mecánico	19
III. 1. 1. Conceptos Preliminares	19
III. 1. 2. Diseño modular.....	21
III. 1. 2. 1. Plataforma móvil.....	21
III. 1. 2. 2. Brazo manipulador	26
III. 1. 2. 3. Efector final.....	33
III. 2. Arquitectura.....	35
III. 2. 1. Lista de componentes, materiales y piezas.	38
III. 3. Consideraciones para manufactura.	40
IV. MODELADO CINEMÁTICO.....	42
IV. 1. Manipulador serial.	42
IV. 1. 1. Cinemática directa.....	42

IV. 1. 1. 1. Notación de Denavit – Hartenberg.	42
IV. 1. 1. 2. Interpretación geométrica de manipulador.....	45
IV. 1. 2. Cinemática inversa.	46
IV. 1. 2. 1. Problema de posición.	47
IV. 1. 2. 2. Problema de orientación.	51
IV. 2. Plataforma móvil.....	54
IV. 2. 1. Propagación de velocidades 55	55
IV. 2. 2. Restricciones cinemáticas. 57	57
IV. 2. 3. Coordenadas generalizadas de configuración. 58	58
IV. 3. Acoplamiento.	59
V. COORDINACIÓN DE MOVIMIENTOS.....	61
V. 1. Planificación de la trayectoria.	61
V. 1. 1. Campos potenciales 61	61
V. 1. 2. Docking y pre-docking de la planeación 63	63
V. 2. Operación <i>pick and place</i>.	65
V. 2. 1. Perfil de trayectoria..... 65	65
V. 2. 2. Determinación del lugar geométrico 67	67
V. 2. 3. Orientación del efector final..... 70	70
V. 3. Algoritmo de coordinación.	72
V. 3. 1. Configuración preferida..... 72	72
V. 3. 2. Definición del punto de referencia 75	75
V. 3. 3. Algoritmo..... 76	76
V. 4. Diseño del experimento en <i>Simulink</i>.	78
V. 5. Reactivision.	86
V. 6. Comunicación computadora-manipulador móvil.	89
V. 7. Comunicación Plataforma móvil con Arduino BT mediante protocolo I²C.	93
VI. EXPERIMENTACIÓN Y RESULTADOS.....	96
VI. 1. Ambiente de experimentación.....	96
VI. 1. 1. Entorno de prueba. 96	96
VI. 1. 2. Equipo..... 96	96
VI. 2. Descripción del experimento.	98
VI. 3. Adquisición de datos.	102

VI. 3. 1. Espacio articular.	104
VI. 3. 2. Espacio cartesiano.	110
VI. 3. 2. 1. Plano x-y.	111
VI. 3. 2. 2. Posturas de brazo manipulador y plataforma móvil.	112
VI. 3. 2. 3. Planeación de trayectoria y lugar geométrico.	113
VI. 4. Análisis de resultados.	117
VI. 4. 1. Espacio articular.	117
VI. 4. 2. Espacio cartesiano.	121
VII. CONCLUSIONES Y TRABAJO FUTURO.	126
VIII. APÉNDICES.	132
VIII. 1. Vision por medio de reactivision.	132
VIII. 1. 1. Cliente TUIO.	132
VIII. 1. 2. Código para adquisición de datos por reactivision en función S.	132
VIII. 1. 3. Ajuste de datos de visión.	135
VIII. 2. Código de arduino.	136
VIII. 3. Tarjeta md25.	139
VIII. 4. Código de funciones S.	139
VIII. 4. 1. Etapa de selección.	139
VIII. 4. 2. Etapa de procesamiento.	140
VIII. 4. 2. 1. Movimiento del brazo manipulador.	140
VIII. 4. 2. 2. Cinemática inversa del brazo manipulador.	142
VIII. 4. 2. 3. Acoplamiento.	143
VIII. 4. 2. 4. Llegando a la meta.	144
VIII. 4. 2. 5. Restricciones cinemáticas de la plataforma móvil.	145
VIII. 5. Diagramas de bloques de Simulink.	145
VIII. 5. 1. Programa en Simulink.	145
VIII. 5. 2. Etapa de ejecución.	149
VIII. 5. 2. 1. Ajuste de los datos del brazo manipulador.	149
VIII. 5. 2. 2. Ajuste de los datos de la plataforma móvil.	149
IX. REFERENCIAS.	150

LISTA DE TABLAS Y FIGURAS

<i>Figura I – 1. Principales tipos de juntas usadas en manipuladores.....</i>	<i>1</i>
<i>Figura I – 2 Los cuatro tipos básicos de ruedas: (a) Rueda Fija. (b) Rueda de Castor. (c) Rueda Sueca. (d) Rueda esférica.....</i>	<i>3</i>
<i>Figura II - 1 Cognitive ServiceRobot.....</i>	<i>8</i>
<i>Figura II – 2 ARMARIII</i>	<i>8</i>
<i>Figura II – 3 Meka M1.....</i>	<i>9</i>
<i>Figura II – 4 PR2 de Willow Garage.....</i>	<i>9</i>
<i>Figura II – 5 Home Exploring Robotic Butler (HERB).....</i>	<i>10</i>
<i>Figura II – 6 Little Helper</i>	<i>10</i>
<i>Figura II – 7 Manipulador Móvil MM – 500.....</i>	<i>11</i>
<i>Figura II – 8 Manipulador Móvil KUKA youBot</i>	<i>11</i>
<i>Figura II – 9 Diagrama de flujo de información para enfoque de redes neuronales.....</i>	<i>12</i>
<i>Figura II – 10 La plataforma (B) y el brazo (A) con sus trayectorias y puntos de referencia</i>	<i>15</i>
<i>Figura II – 11 Manipulador Móvil Seleccionado.....</i>	<i>17</i>
<i>Figura II – 12 Esquema de la Tarea a realizar por el Manipulador Móvil</i>	<i>18</i>
<i>Figura III – 1. Motores, soportes, llantas y tarjeta controladora utilizada para la plataforma móvil.....</i>	<i>20</i>
<i>Figura III – 2. Servomotores de Aeromodelismo a utilizar</i>	<i>20</i>
<i>Figura III – 3. Forma y dimensiones de la placa de transporte</i>	<i>22</i>
<i>Figura III – 4. Medidas de la pila utilizada.....</i>	<i>23</i>
<i>Figura III – 5. Modelo en Nx7 de la placa base y sus componentes.....</i>	<i>23</i>
<i>Figura III – 6. Placa de apoyo (a) y modelo de la rueda loca utilizada (b).....</i>	<i>24</i>
<i>Figura III – 7. Dos diferentes versiones de la plataforma móvil: a) Con dos placas de transporte, b) Con una placa de transporte</i>	<i>25</i>
<i>Figura III – 8. Modelo de arquitectura tipo PUMA.....</i>	<i>26</i>
<i>Figura III – 9. Diagrama de momentos para las juntas del manipulador</i>	<i>27</i>
<i>Figura III – 10. Consideraciones previas: a) Ensamble espiga y caja, b) Sujeción de servomotores.....</i>	<i>28</i>
<i>Figura III – 11. Piezas que forman la base del manipulador: a) pza. de movimiento, b) pza. de sujeción, c) pza. base.....</i>	<i>29</i>
<i>Figura III – 12. Piezas ensambladas que forman junta del hombro y el segundo eslabón</i>	<i>30</i>
<i>Figura III – 13. Conexión entre eslabón dos y eslabón tres.....</i>	<i>31</i>
<i>Figura III – 14. Piezas que forman al eslabón tres y los componentes que contiene</i>	<i>32</i>
<i>Figura III – 15. Modelo en NX7 del brazo manipulador utilizado</i>	<i>32</i>
<i>Figura III – 16. Piezas que forman la estructura principal del gripper.....</i>	<i>34</i>
<i>Tabla III – 1. Tabla de correspondencia para abrir y cerrar gripper.....</i>	<i>34</i>
<i>Figura III – 17. Mecanismo biela manivela y corredera encargado de abrir y cerrar gripper</i>	<i>35</i>
<i>Figura III – 18. Modelo en NX7 del efector final utilizado.....</i>	<i>35</i>

<i>Figura III – 19. Interfaces entre módulos: a) plataforma móvil, b) brazo manipulador y c) efector final</i>	37
<i>Figura III – 20. Manipulador Móvil diseñado en NX7</i>	38
<i>Tabla III – 2. Lista de componentes, materiales y piezas que conforman al Manipulador Móvil diseñando</i>	38
<i>Figura III – 21. Prototipo físico del Manipulador Móvil diseñado</i>	40
<i>Figura IV – 1. Notación de Denavit – Hartenberg para manipulador diseñado</i>	44
<i>Tabla IV – 1. Parámetros de Denavit –Hartenberg para manipulador diseñado</i>	44
<i>Figura IV – 2. Solución para la primera junta</i>	48
<i>Figura IV – 3. Solución para la segunda junta</i>	49
<i>Figura IV – 4. Solución para la tercera junta</i>	51
<i>Figura IV – 5. Cálculo de la coordenada z del punto C del manipulador</i>	52
<i>Figura IV – 6. Solución para la cuarta junta</i>	53
<i>Figura IV – 7. Postura de la Plataforma móvil</i>	55
<i>Figura IV – 8. Vectores de velocidad de los eslabones colindantes [19]</i>	56
<i>Figura IV – 9. Punto de referencia que determina el acoplamiento entre ambas partes</i>	60
<i>Figura V – 1. Llegando a la meta</i>	62
<i>Figura V – 2. Planeación de trayectoria</i>	64
<i>Figura V – 3. Polinomio de interpolación 3-4-5</i>	67
<i>Figura V – 4. Transformación de coordenadas globales a coordenadas del manipulador</i>	70
<i>Figura V – 5. Determinación del ángulo de la última junta del brazo manipulador</i>	71
<i>Figura V – 6. Índice de Manipulabilidad</i>	74
<i>Figura V – 7. Configuración preferida</i>	75
<i>Figura V – 8. Flujo de información</i>	80
<i>Figura V – 9. Etapa de decisión o selección</i>	82
<i>Figura V – 10. Etapa de procesamiento. Bloques de brazo manipulador</i>	83
<i>Figura V – 11. Etapa de procesamiento. Bloque de acoplamiento</i>	84
<i>Figura V – 12. Etapa de procesamiento. Bloques de la plataforma móvil</i>	85
<i>Figura V – 13. Representación gráfica</i>	86
<i>Figura V – 14. Ejemplos de identificadores Fiducials</i>	87
<i>Figura V – 15. Detección de Fiducials</i>	87
<i>Figura V – 16. Etapa de adquisición</i>	88
<i>Figura V – 17. Traslado de puntos de detección: a) para objetos y destinos b) para manipulador móvil</i>	89
<i>Figura V – 18. Arduino BT</i>	90
<i>Figura V – 19. Esquema de comunicación computadora – manipulador móvil</i>	91
<i>Figura V – 20. Etapa de ejecución</i>	93
<i>Figura V – 21. Tarjeta MD25</i>	94
<i>Figura V – 22. Correspondencia de velocidad angular de llantas y velocidad de MD25</i>	95
<i>Figura VI – 1. Ambiente de experimentación</i>	97
<i>Figura VI – 2. Interfaz de reacTIVision</i>	97

<i>Figura VI – 3. Interfaz de usuario.....</i>	<i>98</i>
<i>Figura VI – 4. Proyección del punto del MM al piso</i>	<i>99</i>
<i>Figura VI – 5. Espacio de juntas para: Recogiendo objeto uno, a) Junta Base de manipulador, b) Junta Hombro de manipulador, c) Junta Codo de manipulador, d) Junta Inclinación de Muñeca, e) Junta Giro de Muñeca, f) Abrir y cerrar del gripper, g) Velocidad en rpm de llanta derecha de la plataforma móvil, h) Velocidad en rpm de llanta izquierda de la plataforma móvil</i>	<i>104</i>
<i>Figura VI – 6. Espacio de juntas para: Colocando objeto uno, a) Junta Base de manipulador, b) Junta Hombro de manipulador, c) Junta Codo de manipulador, d) Junta Inclinación de Muñeca, e) Junta Giro de Muñeca, f) Abrir y cerrar del gripper, g) Velocidad en rpm de llanta derecha de la plataforma móvil, h) Velocidad en rpm de llanta izquierda de la plataforma móvil</i>	<i>106</i>
<i>Figura VI – 7. Espacio de juntas para: Recogiendo objeto dos, a) Junta Base de manipulador, b) Junta Hombro de manipulador, c) Junta Codo de manipulador, d) Junta Inclinación de Muñeca, e) Junta Giro de Muñeca, f) Abrir y cerrar del gripper, g) Velocidad en rpm de llanta derecha de la plataforma móvil, h) Velocidad en rpm de llanta izquierda de la plataforma móvil</i>	<i>107</i>
<i>Figura VI – 8. Espacio de juntas para: Colocando objeto dos, a) Junta Base de manipulador, b) Junta Hombro de manipulador, c) Junta Codo de manipulador, d) Junta Inclinación de Muñeca, e) Junta Giro de Muñeca, f) Abrir y cerrar del gripper, g) Velocidad en rpm de llanta derecha de la plataforma móvil, h) Velocidad en rpm de llanta izquierda de la plataforma móvil</i>	<i>109</i>
<i>Figura VI – 9. Posición del punto P de la plataforma móvil en el plano x-y</i>	<i>111</i>
<i>Figura VI – 10. Posición del punto p del brazo manipulador en el plano x-y</i>	<i>111</i>
<i>Figura VI – 11. Postura de la plataforma móvil.....</i>	<i>112</i>
<i>Figura VI – 12. Posición del efector final en el espacio cartesiano.....</i>	<i>112</i>
<i>Figura VI – 13. Velocidades lineales y angulares resultado la planeación de la trayectoria para la plataforma móvil: a) Recogiendo al objeto uno, b) Colocando al objeto uno, c) Recogiendo al objeto dos, d) Colocando al objeto dos</i>	<i>114</i>
<i>Figura VI – 14. Lugar geométrico descrito por la trayectoria del brazo manipulador: a) Recogiendo objeto uno, b) Colocando objeto uno, c) Recogiendo objeto dos, d) Colocando objeto dos</i>	<i>115</i>

I. INTRODUCCIÓN

Dentro de la amplia variedad de sistemas robóticos existente hoy en día, para fines de este trabajo, se distinguen dos de gran importancia: la Robótica Serial y la Robótica Móvil. La Robótica Serial o también conocida como Manipulación Serial es aquella donde el robot está formado de una cadena de eslabones conectados en serie a través de juntas de revolución o prismáticas, entre otras (Figura I – 1). Estas cadenas se dicen abiertas debido a que el primer eslabón se encuentra fijo a una referencia y el último está abierto al exterior, es decir, sólo está unido por el eslabón anterior y la posición de este último se define por la de los elementos anteriores, al final de la cadena se encuentra un efector final, que puede ser una herramienta para realizar alguna tarea. Sus principales aplicaciones son de carácter industrial y la mayoría de los manipuladores industriales tienen al menos seis grados de libertad (juntas prismáticas, de revolución, entre otras), de los cuales los primeros sirven para posicionar al Efector Final (EE) y los últimos para darle orientación.

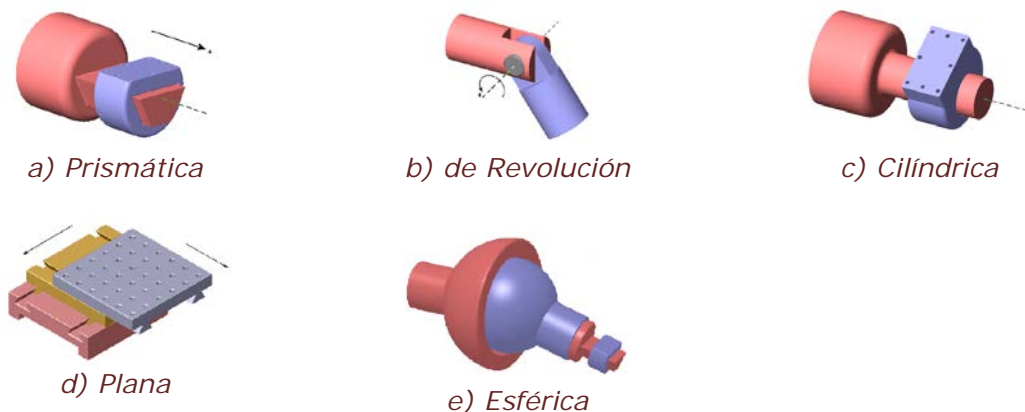


Figura I – 1. Principales tipos de juntas usadas en manipuladores.

Debido a lo anterior, un manipulador debe ser reprogramable y multifuncional, para ello se deben de resolver las siguientes problemáticas:

- Configuración del manipulador.
- Trayectoria a seguir.
- Orientación del órgano terminal o Efector Final.
- Evolución de las juntas, entre otras.

Para poder encontrar respuesta a estos problemas es necesario conocer su cinemática. En robótica son dos tipos: la cinemática directa y la cinemática inversa. La cinemática directa lleva de manera directa a través de la posición de cada junta a la posición global del Efector Final (del espacio articular al espacio

cartesiano), mientras que la cinemática inversa consiste en encontrar la posición de las juntas empleando la posición del Efecto Final en coordenadas globales (del espacio cartesiano al espacio articular).

La dinámica se ocupa de la relación entre las fuerzas que actúan sobre un cuerpo y el movimiento que en él se originan. Por lo tanto, el modelo dinámico de un robot tiene por objetivo conocer la relación entre el movimiento del robot y las fuerzas aplicadas en el mismo. Si se requiere que el manipulador sea más preciso, que llegue a la posición de cierta manera, realizar movimientos rápidos involucrando aceleraciones y masas importantes, entonces es necesario obtener el modelo dinámico del mismo.

Por otra parte, se encuentra la robótica móvil, que se encarga de estudiar aquellos dispositivos capaces de moverse a través de un ambiente o terreno dado, que bien puede ser estructurado o no estructurado. Estos dispositivos pueden ejecutar varias operaciones de acuerdo a su técnica de navegación programada, considerando el ambiente en el que se encuentren. De ahí que, haya robots móviles del tipo aéreo, submarino, locomoción por patas, locomoción por ruedas. Estos últimos siendo los más estudiados por ser la forma más utilizada por el hombre para conseguir movilidad en el medio terrestre, así como por las velocidades que pueden alcanzar. En estos sistemas es muy importante otorgarles la mayor autonomía posible, entendiendo por autonomía como la capacidad de actuar completamente independiente de cualquier entrada humana. El hecho de decir que un robot es autónomo, significa que es capaz de autocontrolarse y autoalimentarse. El estudio de los robots móviles está principalmente concentrado en: cómo moverse de un lado a otro en un ambiente conocido o desconocido. Lo anterior involucra principalmente tres problemas a resolver:

- a) La planificación: dentro de ésta se incluye la planificación del camino y de la trayectoria. En este mismo punto se considera: la evasión de obstáculos y la planeación de la velocidad, entre otras.
- b) La percepción: localizar obstáculos, estimar la posición del robot móvil, construir su entorno.
- c) El control.

Para poder resolver lo anterior mencionado es necesario obtener el modelo cinemático del robot móvil, al igual que con los manipuladores, sólo que para estos no es tan simple, quedando sujeto al tipo de móvil que se desee utilizar. Cada robot móvil cuenta con diferentes tipos de movimiento y esto principalmente al tipo de ruedas que utilizan, como las que se muestran en la Figura I-2. De acuerdo a la configuración de las llantas se obtiene el tipo de robot móvil, estos robots en algunos casos cae en que las restricciones cinemáticas que se obtienen son no-holónicas, es decir, los grados de libertad controlables son menos que el número total de grados de libertad del sistema. Por ejemplo, un automóvil sólo puede navegar atrás, adelante y cambiando dirección pero no puede avanzar

lateralmente, haciéndolo más difícil que un manipulador, ya que no posee este tipo de restricciones. La forma de resolver esta problemática es estudiando las restricciones cinemáticas del robot móvil, modelos de planeación, de percepción, entre otros.

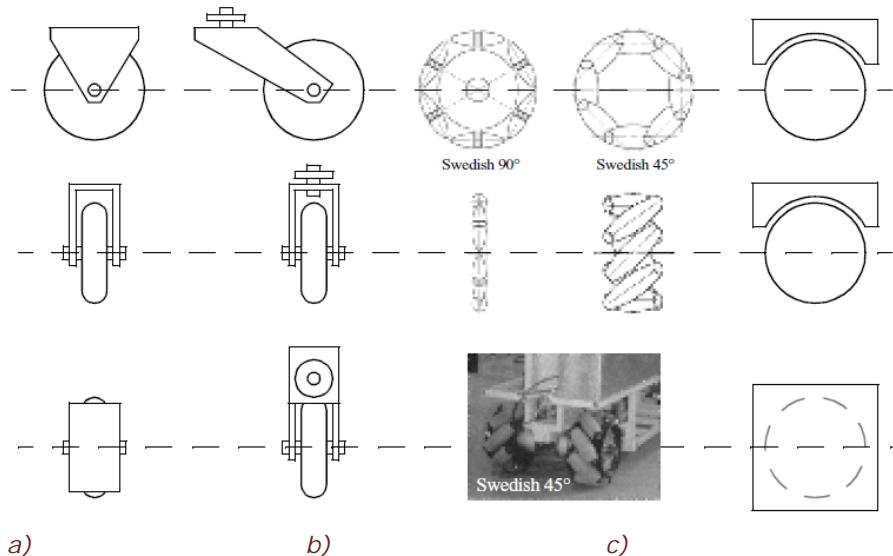


Figura 1 – 2 Los cuatro tipos básicos de ruedas: (a) Rueda Fija. (b) Rueda de Castor. (c) Rueda Sueca. (d) Rueda esférica.

De lo anterior, se deduce que, dependiendo del robot con el que se trabaje, las aplicaciones que tenga serán diferentes. De las dos anteriores mencionadas sus aplicaciones son diferentes, por ejemplo, en un manipulador como su nombre lo dice “manipula” de forma que no puede desplazar su cuerpo a otro sitio, debido a que su referencia u “origen” es fijo, al igual que un robot móvil no puede tomar un objeto y llevarlo a otra posición, de ahí que haya surgido la necesidad de crear otra rama la cual se llama robótica híbrida. Como el nombre lo dice es la unión de dos o más ramas (diferentes) de la robótica. Teniendo esta rama de estudio, las posibilidades de realizar tareas más complejas aumentan, sólo se necesita hacer la combinación adecuada.

Cuando se combinan más de un tipo de robot, se puede considerar que es robótica híbrida, por ejemplo si se une un robot serial con un robot móvil. Con lo que se puede hablar de la **Manipulación Móvil**. Al realizar esta combinación, lo que se obtiene es la suma de las problemáticas de ambas partes, siendo este caso, que los problemas del manipulador serial tienen que resolverse, así como también se tienen que resolver los problemas de la plataforma móvil. Aun cuando se tienen las dos partes solucionadas, surge una nueva serie de problemas, que por sí solas, no las tendrían, sino son resultado de formar la manipulación móvil. Dentro de estos problemas se encuentra:

- 1) Definir la arquitectura del Manipulador Móvil, es decir, dónde se situará cada parte del sistema.

- 2) Definir la tarea, qué es lo que el robot va a hacer.
- 3) Cómo **coordinar** ambas partes, esto es, decidir qué parte se va a mover y en qué momento, incluyendo cómo acoplarlas: la regla que los mantendrá unidas. Esta problemática ha atraído la atención de investigadores en la actualidad.

James Tsay et ál. [3], en el manipulador tienen una cámara montada en el efector final que guía al robot, la plataforma móvil tiene dos cámaras que le hacen posible deformar su trayectoria, en este momento, se enfocan más en llegar a la posición que recoger el objeto. La forma en que coordinan no es su tema principal, sin embargo, Yingshu Chen et ál. [1] modelan el brazo mediante transformaciones homogéneas, mientras que en la plataforma móvil toman la posición inicial como referencia y calculan el movimiento siguiente de acuerdo a una posición posterior. Este movimiento, depende de la dirección a donde la plataforma vaya, consideran ambos casos (horario y anti-horario); para acoplar ambas partes las unen mediante la transformación de coordenadas de la base del manipulador a la orientación del móvil. El control lo hacen mediante una red neuronal en tres niveles: decisión, procesamiento, ejecución. La coordinación la hacen mediante dos sub-controladores (no indicados) uno para cada parte del Manipulador Móvil, donde la red neuronal toma la decisión. Chungyan Gao et ál. [5] consideran al manipulador móvil unido, es decir, como un solo sistema, trasladan el origen del manipulador al marco de referencia de la plataforma móvil y trabajan en coordenadas globales. Al igual que en [1] utilizan redes neuronales para la toma de decisiones basadas en un algoritmo difuso, Gilles Foulon et ál. [2], al igual, tratan ambas partes como una sola. El marco de referencia del manipulador es el mismo de la plataforma móvil. Utilizan el espacio general de coordenadas (posición en x, posición en y, orientación de la plataforma, juntas del manipulador) donde la salida de su control son las coordenadas globales del efector final y su orientación. Logran unir al sistema mediante el jacobiano de ambas partes y de esta forma coordinan sus movimiento. M. Egerstedt y X. Hu [4] a diferencia de los anteriores, consideran a las partes separadas y dan la trayectoria que debe seguir el efector final, respecto a esto planean la trayectoria de la base, de manera que el efector final siempre se mantenga en el espacio de trabajo, esta trayectoria la trazan mediante el enfoque de vehículo virtual con una distancia "look ahead", realizan su control de manera que se pueda aplicar a cualquier robot. Y. Yamamoto y X. Yun [6] trabajan muy similar, solo que ellos no consideran al manipulador en la dinámica. El manipulador se debe de mantener en la posición óptima, el operador arrastra el efector final y la plataforma móvil responde de forma que se mantenga en esta posición en todo momento, al igual consideran ambas partes separadas y las unen utilizando el punto final del manipulador como punto de referencia para el movimiento de la plataforma, este punto estaría siendo

el “*look ahead*” de la plataforma móvil y deberá de mantener al manipulador en la posición óptima.

Como se acaba de mencionar, trabajan al manipulador móvil como un solo sistema haciendo su análisis más complejo. También queda claro que el método de las redes neuronales es lo más utilizado para su planeación de movimientos, además de utilizar el modelo dinámico del sistema para poder controlarlo. Todo lo anterior indica que la utilización de un manipulador móvil es muy compleja además de impráctica, pues resulta ser un tema relativamente nuevo y todavía queda mucho por explorar sobre él. De ahí que se apliquen teorías complejas para justificar el movimiento, lo que hace que se pierda el enfoque práctico. Puede ser más sencillo, sino se consideran teorías complejas, ni metiéndose a la dinámica del sistema; y funcional, es decir, que pueda realizar una tarea, que sea útil. Cabe aclarar, que no se trata de desacreditar o menospreciar el trabajo de los autores mencionados, no obstante se propone trabajarlo con un enfoque más práctico, que potencie la funcionalidad.

El propósito de la presente tesis es estudiar la forma de coordinar los movimientos de un manipulador móvil que se compone de un manipulador serial acoplado a un robot móvil con ruedas, con la finalidad de que ambas partes trabajen en conjunto para cumplir una tarea, de manera sencilla, sin teorías complejas, con un algoritmo de coordinación heurístico, que permita que ambas partes se muevan sin involucrar la dinámica de estas. Con lo anterior, se puede demostrar que es posible abordar este tipo de problemas con más facilidad y así dar pie a más investigadores a realizar tareas prácticas con el fin de que el beneficio a la humanidad sea mayor.

El presente trabajo se desarrolla en siete capítulos. El primer apartado es la introducción, en donde se le dice al lector los temas que abordará esta tesis.

En el capítulo dos, se hablará sobre la manipulación móvil y se expondrá una recopilación del trabajo realizado en esta área, así mismo se presenta el manipulador móvil con el que se trabajó y el diseño de la tarea a realizar.

En el capítulo tres, una vez que se seleccione el manipulador con el que se va a trabajar, se mostrará el diseño, el modelo y la construcción del manipulador móvil. También, se mostrarán todos los elementos que forman al robot.

En el capítulo cuatro, se explicará cómo se obtiene el modelo cinemático del manipulador serial, su cinemática directa e inversa; en lo que se refiere a la plataforma móvil, se plantearán sus restricciones cinemáticas y se indicará la forma de obtenerlas, finalmente se mostrarán las ecuaciones que se utilizaron para acoplar ambas partes y tratarlas como una sola.

El cuerpo del trabajo se desarrolla en el capítulo cinco, aquí se mostrará la planificación de movimientos de la plataforma móvil por medio de campos potenciales, se explicará la operación *pick and place* para el manipulador serial, también se presentará el algoritmo heurístico que se utilizó para coordinar los

movimientos del robot además de la forma en la que se realiza el experimento, es decir, todos los elementos que forman la prueba.

Las pruebas realizadas y su descripción, el ambiente en el que se llevan a cabo y el análisis de los resultados obtenidos del algoritmo heurístico de coordinación, propuesto e implementado en el manipulador móvil elaborado, se presentarán en el capítulo seis.

Para finalizar en el capítulo siete, se presentan las conclusiones de las pruebas realizadas y la propuesta del trabajo futuro a realizar.

Al final del documento, el lector encontrará las referencias y los apéndices que sirvieron a la justificación de esta tesis.

II. MANIPULACIÓN MÓVIL

En el presente trabajo se refiere a Manipulador Móvil como una plataforma móvil (con ruedas) con un manipulador serial montado. En [7] y [8] se investiga sobre otro tipo de manipuladores móviles.

Un Manipulador Móvil, es un sistema mecánicamente unido comprende de una plataforma moviéndose, sujeta a sus restricciones cinemáticas, y a los grados de libertad del brazo manipulador montado en ella. Estos sistemas combinan las ventajas de las plataformas móviles y los brazos robóticos, además, reducen sus desventajas. Por ejemplo, un manipulador móvil tiene más amplio espacio de trabajo cuando posee una plataforma móvil, pues ofrece mayor funcionalidad durante la operación.

Hoy en día la planeación de movimientos y el control de estos sistemas han llamado mucho la atención por su utilidad. El control de los manipuladores móviles principalmente contempla dos aspectos: la planeación de movimiento, tanto para la plataforma móvil como para el manipulador, y la otra, el control coordinado de ambos. Una vez que se tienen estos dos primero, lo siguiente es asignar la tarea. En muchas ocasiones movilizar al manipulador resulta en redundancia cinemática, esto se puede solucionar coordinando la plataforma y al manipulador. El tipo de coordinación a utilizar dependerá de las especificaciones de la tarea.

Algunas de las aplicaciones más importantes de los manipuladores móviles son: desempeñar tareas que involucren explosivos, ambientes peligrosos como exploración de exteriores, operaciones espaciales, construcción, tareas con fines militares y servicio personal.

II. 1. TIPOS DE MANIPULADORES MÓVILES

Los manipuladores móviles se pueden dividir por tipo de manipulador y tipo de plataforma móvil. El manipulador puede ser humanoide o bien un brazo manipulador. A su vez la plataforma móvil puede ser omnidireccional, diferencial, etc. entre otros tipos de locomoción. A continuación, se muestran algunos manipuladores móviles con los que hoy en día se trabaja alrededor del mundo. Con el fin de realizar una topología de estos, se dividirán por el tipo de manipulador que está montado sobre la plataforma móvil: Humanoide o Brazo Manipulador.

II. 1. 1. MANIPULADOR MÓVIL DE MANIPULADOR HUMANOIDE.

Cognitive Service Robot (Cosero) [9]: Desarrollado por grupo de Sistemas

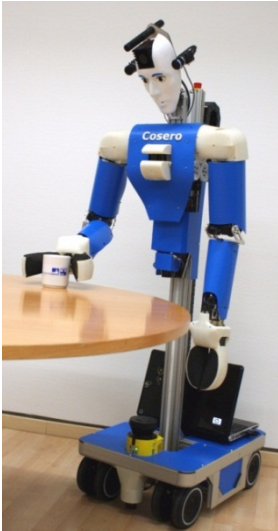


Figura II - 1 Cognitive Service Robot

Autónomos Inteligentes en la Universidad de Bonn, Figura II – 1. Realizado para tareas de manipulación móvil en ambientes domésticos. La locomoción está dada por una plataforma móvil omnidireccional, que cuenta con cuatro pares de llantas actuadas. La parte del manipulador es humanoide, que consiste de dos brazos de siete grados de libertad (GDL), equipado con un *gripper* de dos dedos y una cabeza con movimiento en el eje horizontal y en el eje vertical (*pan & tilt*). El cuerpo superior puede moverse verticalmente, ampliando el espacio de trabajo de los brazos.



Figura II – 2 ARMAR III

ARMAR –III [9]: Cuenta con 43 GDL, siete en la cabeza, siete en cada brazo, ocho en cada mano, tres en el torso y tres en la plataforma móvil, Figura II – 2. La cabeza cuenta con dos ojos, cada ojo se puede mover independientemente. A su vez cada ojo cuenta con dos cámaras a color, una para la visión periférica y la otra para la foveal. El sistema de visión se encuentra montado en mecanismo de cuatro GDL. Para la localización acústica cuenta con un arreglo de seis micrófonos y además un sensor inercial instalado en la cabeza para estabilizar las imágenes de la cámara. La locomoción del robot es realizada por una plataforma móvil omnidireccional, esta cuenta con una combinación de tres láseres y encoders ópticos.

Meka M1 [10]: Cuenta con dos brazos manipuladores ligeros de siete GDL, con control por fuerza. Diseñados para coincidir con el tamaño y forma del brazo de un adulto joven, Figura II – 3. Un par de manos con cinco GDL, aproximadamente del tamaño de un humano. La cabeza cuenta con siete GDL para visión activa y una cámara de alta definición en cada ojo. Para lograr la locomoción tiene una plataforma móvil omnidireccional, al momento de adquirirlo se incluye el software para utilizarlo.

PR2 de *Willow Garage* [11]: Este manipulador móvil (Figura II – 4), por la parte de la manipulación cuenta con dos brazos controlados por corriente, de forma que pueden ser utilizados en ambientes estructurados. Cuenta con un resorte pasivo que hace contra balance permitiendo, incluso, cuando no se encuentre energizado, que los brazos se encuentren suspendidos; cada brazo cuenta con cuatro GDL. Cada muñeca tiene dos GDL de alto torque, que le permite manipular cualquier tipo de objetos, y su *gripper* de un GDL. Para la

movilidad del robot tiene una espina telescópica. La locomoción la realiza mediante una plataforma móvil omnidireccional, lo que permite que PR2 pueda trabajar en cualquier espacio. De igual forma que en los anteriores mencionados, la cabeza tiene movimiento en el eje horizontal y el vertical, además de una infinidad de sensores: cámaras en la cabeza, escáner láser en los hombros, cámara en los antebrazos, sensores de presión en el *gripper*. Finalmente la plataforma cuenta con un escáner láser, entre otras cosas que ofrece.



Figura II – 3 Meka M1

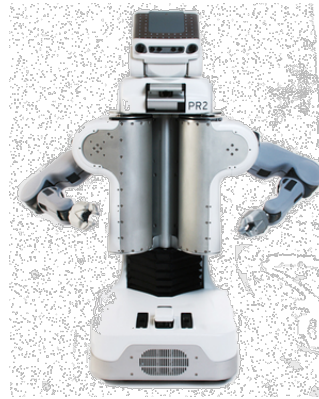


Figura II – 4 PR2 de Willow Garage

II. 1. 2. MANIPULADOR MÓVIL DE BRAZO MANIPULADOR

HERB (Home Exploring Robotic Butler) [12]: Manipulador móvil construido por *Intel Research Pittsburg* (Figura II – 5), en colaboración con el Instituto de Robótica de la Universidad de Canegie Mellon. Cuenta con un *Segway* como plataforma móvil, un brazo robótico de *Barrett Technology* con siete GDL y cinemática parecida a la humana. Dentro del robot, un par de computadoras de bajo consumo y la comunicación se realiza por medio de redes inalámbricas. En la parte de los sensores cuenta con dos cámaras en el hombro del robot, una para la localización de marcadores y la otra para el reconocimiento de objetos, éstas, con solo el movimiento en el eje horizontal que proporciona el mismo hombro. En el brazo se encuentra otra cámara para detectar la forma de los objetos y finalmente en la base móvil se encuentra un escáner láser para la localización y el mapeo. Sin contar todo el software que requiere, que en su mayoría, son librerías de uso libre.

Little Helper [13]: Se trata de una investigación actual y proyecto en desarrollo del Departamento de Ingeniería Mecánica y Manufactura, Universidad de Aalborg, Dinamarca (Figura II – 6).



Figura II – 5 Home Exploring Robotic Butler (HERB)



Figura II – 6 Little Helper

El sistema de robótica móvil, consiste de una plataforma móvil no-holonómica. El sistema de manipulación, consiste de un brazo robótico de seis GDL que corre a 230 VAC. A su vez, la herramienta o el efector final es un sistema neumático, por ende, contiene su compresor y su tanque de aire dentro de la plataforma. En la plataforma se encuentra un banco de baterías para su uso, así como su computadora interna de 2.0 GHZ *dual core* y 2 GB de RAM corriendo en *Windows XP*. La plataforma móvil, para la seguridad y navegación, tiene un escáner laser y sensores ultrasónicos. La visión consiste de una cámara monocromática *FireWire* y lente ajustable de forma digital además de propio su sistema de iluminación. Entre el software utilizado se encuentran *MATLAB*, *LabVIEW*, y los programas para controlar el robot manipulador y la plataforma móvil.

Manipulador Móvil MM – 500 [14]: Desarrollado por *Neobotix* (Figura II – 7), una empresa que manufactura robots móviles y sistemas robóticos. Poseen una variada cantidad de plataformas y manipuladores, como un solo sistema se encuentra este modelo. La plataforma MP – 500 es no-holonómica. El manipulador *Schunk LWA 3*, consiste de siete GDL, y su *gripper* cuenta con dos dedos paralelos actuados eléctricamente. Dentro de la plataforma se encuentra la batería que lo alimenta y su computadora; se comunica de forma inalámbrica. Cuenta con un escáner laser para la navegación y cinco sensores ultrasónicos, a diferencia de los anteriores manipuladores móviles, éste no cuenta con cámara, solo incluye su software basado en Linux para controlar ambas partes.

Manipulador Móvil KUKA *youBot* [15], [16]: Una empresa que se concentra en soluciones avanzadas para la automatización y procesos industriales. Este manipulador móvil (Figura II – 8), de reciente desarrollo, consiste de una plataforma móvil omnidireccional con cuatro ruedas patentadas por la misma empresa. El manipulador formado por una cadena serial de cinco GDL y un *gripper* de 2 dedos paralelos. Ambas partes alimentadas eléctricamente. Integrado a la plataforma se encuentra su computadora y la alimentación, además, la comunicación se hace vía inalámbrica. Como se trata de un robot para investigación y educación no cuenta con sensores. La paquetería de software con la que cuenta se basa en Linux y librerías de código abierto.



Figura II – 7 Manipulador Móvil
MM – 500



Figura II – 8 Manipulador Móvil KUKA youBot

II. 2. ENFOQUES SOBRE COORDINACIÓN DE MOVIMIENTOS

En la actualidad la coordinación de un Manipulador Móvil es un tema que continua estudiándose. Por lo que no es tan sencillo realizar una topología de las diferentes formas en las que se coordinan los movimientos, por esto, en esta sección se presentan los enfoques más utilizados, en cuanto a coordinación se trata.

Entiéndase como coordinación, a la manera en la que se realizan los movimientos. Un manipulador móvil, como ya se indicó consta de dos partes (la parte de manipulación y la parte de la locomoción), cada parte tiene características diferentes, de forma que se debe diseñar una manera de hacer que ambas partes trabajen como una misma.

II. 2. 1. ENFOQUE DE REDES NEURONALES [1].

En los siguientes párrafos se describe un algoritmo basado en una red neuronal y desarrollado para la coordinación de movimientos del manipulador móvil. Este algoritmo está formado por tres niveles: nivel de decisión, nivel de procesamiento y nivel de ejecución. El algoritmo imita el comportamiento humano con el fin de volver al robot parcial o totalmente autónomo. La idea principal es identificar y compensar la dinámica de ambas partes, así como la interacción de ambas mediante una red neuronal. En la Figura II – 9 se muestra un diagrama de bloques que describe el flujo de información para el enfoque de redes neuronales.

Nivel de decisión. En este nivel se encuentran dos casos: entorno desconocido y entorno conocido. Si el Manipulador Móvil se mueve en un entorno conocido, las trayectorias de movimiento deseadas pueden ser planeadas por este nivel fuera de línea, de manera que, los datos que entregan los sensores no son

necesarios. Sin embargo si el entorno es desconocido y cambiante, la decisión se tiene que hacer en línea, ya que el Manipulador Móvil debe responder de manera pronta a los cambios de su entorno. Por lo que, el nivel de decisión, debe razonar un plan para el Manipulador Móvil, usando ciertos mecanismos de razonamiento y planificación de caminos. Las decisiones son tomadas mediante el conocimiento previo y adquisición de datos de los sensores, como visión, fuerza, contacto, entre otros, que son usados para reconocer el desconocido y cambiante entorno.

Nivel de Procesamiento. En este nivel una red neuronal de función de base radial (RBF), está construida para identificar la dinámica desconocida de ambas partes, así como las interacciones inciertas con el entorno. La entrada de la red neuronal contiene trayectorias de referencia, error de seguimiento de posición y velocidad de ambas partes. La salida de la red neuronal son los términos dinámicos desconocidos, ya identificados de la plataforma y el manipulador.

Al mismo tiempo, almacena las nuevas reglas para actualizar los pesos de la red, en tiempo real, usando ciertas reglas ponderadas de actualización. Las entradas iniciales de la red son números aleatorios y cambian de manera que va aprendiendo. Dentro de la red neuronal RBF, dos servocontroladores están dispuestos para compensar la dinámica de cada parte, los subcontroladores entregan datos de torque para las juntas y las ruedas.

Nivel de ejecución. En este nivel, los pares que son calculados por el nivel de procesamiento, se aplican a los motores de cada junta en el Manipulador Móvil para generar el movimiento de éste. Está compuesto por la parte física, principalmente.

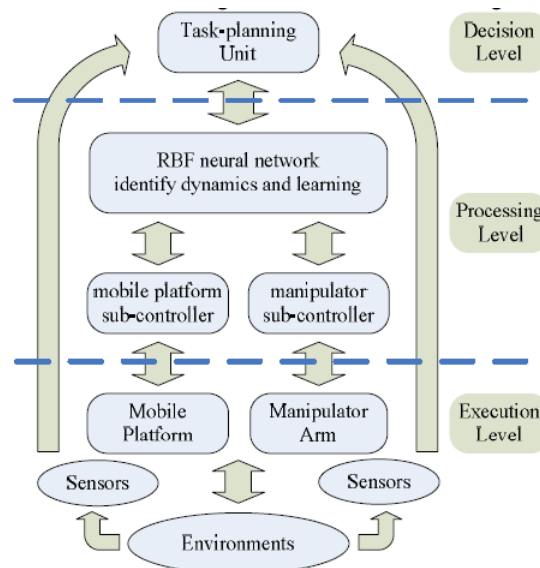


Figura 11 – 9 Diagrama de flujo de información para enfoque de redes neuronales.

II. 2. 2 ENFOQUE CINEMÁTICO DEL JACOBIANO [2].

En este apartado se describe un enfoque cinemático, para coordinar movimientos, mediante el Jacobiano. Para lograr esto primero se introducen algunos conceptos con el fin de dejar más claro este enfoque.

La configuración mecánica de un sistema, es conocida cuando la posición de todos los puntos son conocidos en un marco fijo. Las coordenadas que permiten que se defina la configuración, son llamadas coordenadas generalizadas del sistema mecánico. Este número da la dimensión del espacio generalizado, según el caso del manipulador móvil, son el número de coordenadas generalizadas de la plataforma móvil y las del brazo. Se le llama q al vector de las coordenadas generalizadas. Para describir la localización del EE se tienen las coordenadas operacionales de éste y se denota por el vector x .

Se definen dos tipos de tareas: las generalizadas y las operacionales. Una tarea generalizada a realizar por un sistema mecánico, consiste en llegar a las coordenadas generalizadas y la tarea operacional, consiste en llegar a las coordenadas operacionales. En esta forma de coordinar no se consideran los efectos dinámicos y todas las aportaciones se hacen en un marco cinemático. Se consideran tres tipos de tarea para realizar:

- 1) Tarea generalizada punto a punto (GPTPT): en esta se impone una configuración inicial q^0 y una final q^f , se encuentra el camino en coordenadas generalizadas, uniéndose la configuración inicial a la final.
- 2) Tarea operacional punto a punto (OPTPT): dada q^0 y la localización final x^f , se encuentra el camino operacional x , uniéndose la configuración inicial con la localización final. En este tipo de tarea, mediante el modelo de cinemática inversa, se obtienen las coordenadas generalizadas y así se vuelve una tarea GPTPT. Sin embargo, la cinemática inversa presenta múltiples soluciones, por lo que se recomienda obtener la mejor solución. La mejor solución la encuentran con la maximización del índice de manipulabilidad, haciendo que se encuentre lo más alejado de singularidades.
- 3) Tarea operacional de camino continuo (OCPT): para la localización inicial y final, se encuentran las coordenadas generalizadas para el camino.

Para realizar estos movimientos, es necesario modelar ambas partes del Manipulador Móvil como un solo sistema, referenciados a un marco fijo (global). Al modelo se le aplica el Jacobiano, con el fin de encontrar las singularidades y así evitarlas. De forma que, se obtiene por donde debe moverse la plataforma y los valores que pueden adquirir las juntas. De esta forma, el sistema se mueve a las coordenadas indicadas, de acuerdo al tipo de tarea de las antes mencionadas.

Se realiza el análisis del camino a seguir mediante el Jacobiano. Una vez que se obtienen las zonas por las que puede moverse, se procede a realizar los

movimientos que cada parte del sistema realizará. Se podría decir que la forma en que coordinan los movimientos, es fuera de línea.

II. 2. 3. ENFOQUE DEL VEHÍCULO VIRTUAL [4].

En los siguientes párrafos se describe el enfoque del vehículo virtual, donde la idea principal es que uno defina la trayectoria del EE y con esta calcular los puntos de referencia para la plataforma. Haciendo que ambas partes se muevan de forma coordinada. Al principio se introducen algunos conceptos para su mejor entendimiento.

El movimiento coordinado puede ser visto como dos problemas de seguimiento desacoplados, separando la cinemática del brazo, de la formulación.

El Jacobiano relativo y el Jacobiano del manipulador tienen el mismo determinante, esto quiere decir que, para cualquier movimiento desarrollado por el manipulador estático, se puede realizar por el manipulador con una plataforma móvil. Mientras el EE se mantenga en el espacio de trabajo diestro, el Jacobiano relativo es invertible. Si el manipulador no es redundante, se puede desacoplar el movimiento del brazo y de la plataforma móvil.

La idea general de este enfoque, es como una combinación de seguimiento de trayectoria convencional. Esta trayectoria de referencia está parametrizada en el tiempo. El seguimiento de trayectoria dinámico, tiene como criterio mantenerse cerca del camino geométrico, pero no necesariamente cerca de un punto en específico en el tiempo. Lo anterior quiere decir que, mientras los puntos de referencia se muevan a través de las trayectorias referidas, los robots deben seguirlos con la distancia "*look ahead*" predefinida.

En primera instancia, se diseña la evolución de los dos vehículos virtuales. De esta manera, la distancia deseada proyectada, siempre se encuentra dentro del espacio de trabajo diestro del manipulador.

Para la plataforma móvil, se tienen que encontrar, la velocidad lateral y longitudinal. Para que la plataforma siga al vehículo virtual, debe de ser un camino de referencia fluido. Lo que significa que la velocidad lineal sea diferente de cero. Los objetivos que debe cumplir para la plataforma son:

- La distancia entre el punto de referencia y el centro de la base, debe ser menor o igual a la distancia "*look ahead*" predefinida.
- La diferencia entre el ángulo deseado y el ángulo de la plataforma, debe ser menor igual a un diferencial de ángulo predefinido.

Para realizar el control de la plataforma, se obtiene la velocidad del camino a seguir, en esta formulación de la velocidad, se encuentra un término que sirve para coordinar ambas partes. Este término es calculado con base en la distancia entre los dos puntos deseados. En particular este término asegura que, la velocidad de cambio de la distancia, sea menor a cero si la distancia es mayor a

un valor máximo de la distancia “look ahead”. Este control se encarga de guiar a la plataforma al punto de referencia deseado, con una velocidad proporcional al error, lo que hace que se pueda aplicar a cualquier plataforma móvil.

De forma similar a la plataforma, el efector final debe de seguir a otro vehículo virtual, moviéndose en un camino de referencia en sus tres ejes (x, y, z).

- De igual forma que en el anterior, se requiere que la distancia entre el punto de referencia y el punto del EE, sea menor a la distancia “look ahead” para el EE.

Como en el caso anterior, se obtiene la velocidad del camino a seguir, que es el control para el brazo manipulador. En esta parte se encuentra un término de coordinación. Este término, asegura que la velocidad de cambio entre los puntos deseados sea mayor a cero, siempre y cuando, la distancia es menor al valor mínimo de la distancia “look ahead”. En la Figura II – 10, se muestra un esquema del sistema con sus trayectorias de referencia y los puntos de referencia que se usan.

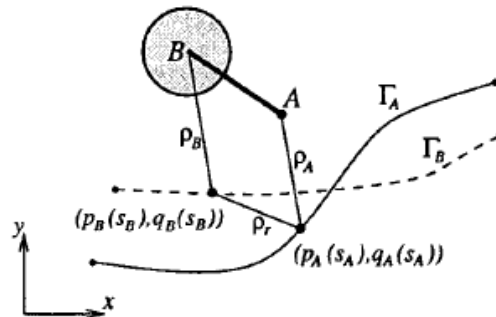


Figura II – 10 La plataforma (B) y el brazo (A) con sus trayectorias y puntos de referencia.

Las ecuaciones dinámicas se encargan de mantener la distancia entre un rango mínimo y máximo definido, es decir, si hay perturbaciones o errores de inicialización, las ecuaciones dinámicas lo ajustan.

II. 2. 4. ENFOQUE DE LA CONFIGURACIÓN PREFERIDA DEL MANIPULADOR [6].

A continuación se describe el enfoque de la configuración preferida del manipulador, en el que la idea principal es: llevar al manipulador a una configuración preferida. La plataforma móvil debe permitir que el manipulador, se mantenga en la configuración preferida.

Es deseable llevar al manipulador a cierta configuración preferida, planeando apropiadamente el movimiento de la plataforma móvil. Si la trayectoria del EE del manipulador es conocida en un sistema de coordenadas, el movimiento de la plataforma móvil puede ser planeado por consecuencia. Si el movimiento del

manipulador no es conocido previamente, la planeación debe realizarse de forma local y en tiempo real.

Lo primero a establecer es, la ecuación de salida que permita realizar la tarea, para esto es conveniente definir un marco coordinado para la plataforma móvil en el centro de masa de la misma. Se escoge un punto respecto al marco coordinado de la plataforma como punto de referencia. La plataforma móvil, se controla de forma, que el punto de referencia siga una trayectoria deseada. La selección de este punto de referencia, tiene como propósito, coordinar la locomoción y la manipulación. El punto final del manipulador, cuando se encuentra en su configuración preferida, se establece como el punto de referencia. Este punto sigue el movimiento que resulta del “arrastre” del operador humano.

Mientras el operador humano arrastra el punto del manipulador, la plataforma móvil deberá llevar al manipulador a la configuración preferida. Se escoge como configuración preferida, la que maximice el índice de manipulabilidad del manipulador. Por lo tanto, la plataforma móvil se desplaza, de tal forma que, el manipulador se mantiene en la configuración preferida.

El índice de manipulabilidad se puede considerar como, la distancia entre la configuración del manipulador y las configuraciones singulares, donde estas últimas vuelven al índice de manipulabilidad cero. En, ó, cerca de las configuraciones singulares, el EE del manipulador, tendrá dificultades en su movimiento, para ciertas direcciones. El objetivo de maximizar el índice de manipulabilidad, es mantener al manipulador lejos de las singularidades. El índice de manipulabilidad se obtiene mediante el determinante del Jacobiano del manipulador, con este se encuentran los valores que lo maximizan.

II. 3. MANIPULADOR MÓVIL SELECCIONADO.

Descrito lo anterior, sigue la selección del tipo de Manipulador Móvil a utilizar, para cumplir con el objetivo.

En un principio, se descartó que la parte del manipulador sea humanoide, debido a su complejidad y a la limitante económica. Estos sistemas llegan a ser de precio elevado, así que, con fines de cumplir con el objetivo: un solo brazo manipulador serial es necesario y suficiente. Este brazo, a su vez está acoplado a una plataforma móvil, igualmente se tuvo que seleccionar.

Las plataformas móviles holonómicas, son las más recomendables para este tipo de sistemas por su gran movilidad. El principal problema de este tipo de plataformas, es la alta complejidad en su manufactura. Por tal motivo, se seleccionó una plataforma móvil no – holonómica, siendo ésta de manufactura más sencilla que la anterior. La plataforma móvil consiste de una configuración diferencial. De acuerdo a lo anterior, puede que el sistema a utilizar sea muy semejante al de la Figura II – 7 de *Neobotix*.

En el siguiente capítulo, se desarrolla el diseño del prototipo que se utilizó a lo largo de este trabajo. En la Figura II – 11, se muestra el resultado del desarrollo del prototipo.

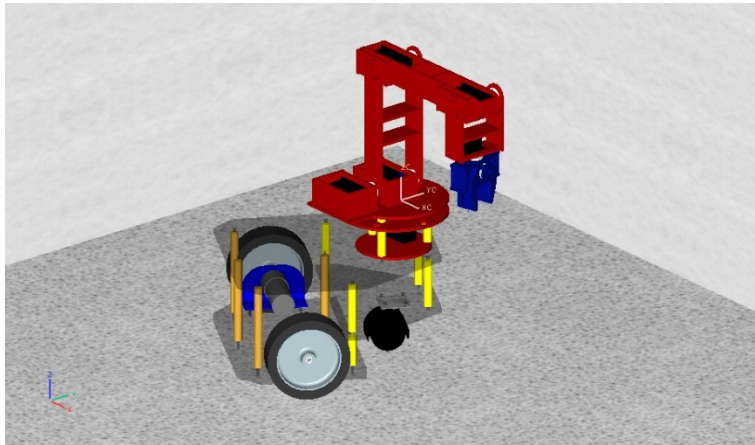


Figura II – 11 Manipulador Móvil Seleccionado.

II. 4. DISEÑO DE LA TAREA A DESEMPEÑAR

Con el fin de demostrar el algoritmo de coordinación planteado y propuesto, se diseña una tarea, que deberá validar el algoritmo. Por lo tanto, a continuación se describe la tarea a desempeñar por el Manipulador Móvil (MM).

En un espacio de trabajo establecido, se tiene al MM y los dos objetos que tiene que recoger. El sistema de visión deberá determinar la ubicación del MM y los objetos. Al principio, identifica qué objeto se encuentra más cerca del MM. Una vez identificado el objeto más cercano, la plataforma móvil se hará cargo de llevar al manipulador al lugar para recoger el objeto **(1)**. Cuando el objeto sea recogido, deberá ser llevado al lugar destinado para su colocación **(2)**. Cuando el objeto es colocado en su destino, el sistema de visión, identifica la ubicación del siguiente objeto. Una vez en la posición indicada, recoge el siguiente objeto **(3)**. Cuando éste sea recogido, se lleva a su lugar destinado **(4)**, que es, diferente al primero. En el momento en que el objeto sea dejado, la prueba termina. Se puede apreciar que esta prueba no está limitada a un cierto número de objetos, es decir, si es necesario se agregan más objetos y sus respectivos destinos. Para fines prácticos, se considera suficiente con dos objetos para validar el funcionamiento del algoritmo. En la Figura II – 12 se ilustra la prueba a desempeñar por el MM.

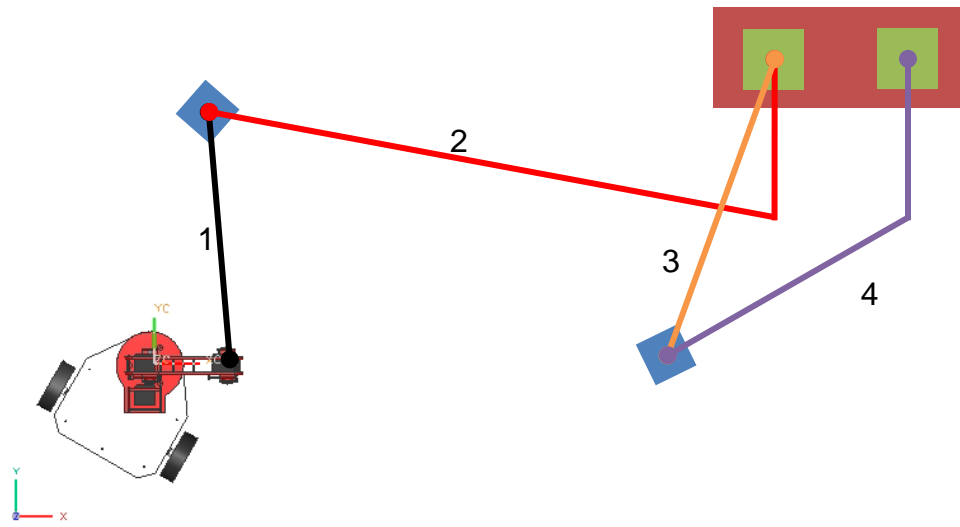


Figura II – 12 Esquema de la Tarea a realizar por el Manipulador Móvil

III. DISEÑO DEL MANIPULADOR MÓVIL

En este capítulo, se describe el proceso por el cual se diseña el prototipo funcional, en el que se implementa el algoritmo de coordinación, analizado en capítulos posteriores. Es muy importante señalar que, no se trata de un producto final, pues el propósito de este trabajo no es el diseño del manipulador móvil, sino un prototipo funcional que permita llegar al objetivo indicado. El modelo que a continuación se desarrolla, es el medio y no el objetivo.

III. 1. DISEÑO MECÁNICO

III. 1. 1. CONCEPTOS PRELIMINARES

El manipulador móvil debe estar formado por tres partes o subsistemas: la plataforma móvil, el brazo manipulador y el efector final. Dependiendo de la tarea que realice el robot, será la configuración y los elementos mecánicos que conformen al mismo.

La plataforma debe cumplir con la función del desplazamiento, debe llevar al manipulador a la posición que se asigna. Por lo tanto, tiene que trazar el camino, que lleve a la plataforma móvil a la postura que permita cumplir con la tarea. Además de proporcionar movimiento, la plataforma debe de contar con espacio para la batería que le proporcionará el voltaje de alimentación, también espacio para colocar todos los componentes que controlan el sistema. El sistema del brazo manipulador, debe posicionar al órgano terminal, en cualquier lugar de su espacio de trabajo. Además el manipulador, tiene que orientar al órgano terminal, para que realice la tarea que se le asigne, es decir, tiene que describir una trayectoria, que lleve cada objeto de una posición inicial a una posición final, permitiendo realizar la tarea. Finalmente el efector final, debe cumplir con la función de agarrar y soltar al objeto. Lo anterior es propio de la tarea a realizar, es decir, si la tarea es diferente a recoger un objeto, se tendrá que utilizar otro tipo de efector final.

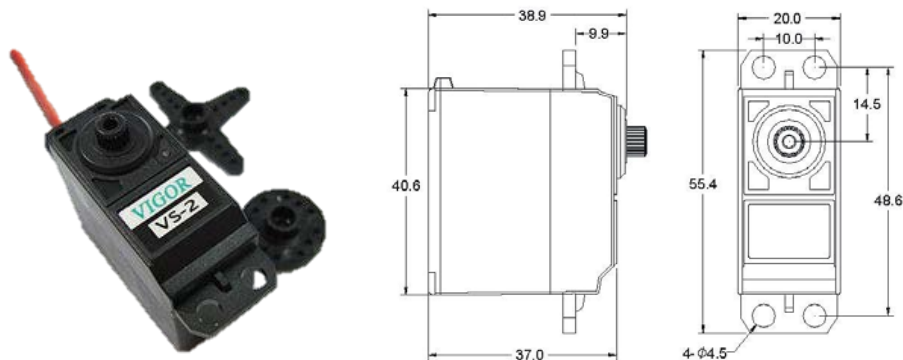
Existen muchas formas de realizar las funciones que se estipulan arriba, por lo que hay que reducir el número de posibles configuraciones. En primera instancia, está la plataforma móvil. Dentro del equipo de trabajo, existe material para realizar prototipos en esta área, por lo que para realizar la plataforma móvil, se utilizó parte del material disponible. El material que se utilizó son: motores de CD, para lograr la locomoción de la plataforma; soportes, llantas y una tarjeta

controladora. Este material se muestra en la Figura III – 1, más adelante se detalla la información sobre este material.



Figura III – 1. Motores, soportes, llantas y tarjeta controladora utilizada para la plataforma móvil.

Para el brazo manipulador y su efector final, se propone utilizar servomotores de aeromodelismo, tamaño estándar, para las juntas del manipulador; y de tamaño mini, para la orientación del efector final y su sistema de sujeción de objeto. Lo anterior lleva a concluir que, no se tratará en este trabajo sobre la dinámica del manipulador. La elección de servomotores de aeromodelismo (Figura III – 2), responde al tamaño que el robot llegaría a tener. Estos servomotores son de un tamaño reducido y un par relativamente alto, la cuestión es que no se puede obtener el par necesario para la tarea, de ahí que no se utilice el modelo dinámico del manipulador. Como ya se mencionó es un prototipo funcional, no el producto final. Otra propuesta, es utilizar acrílico como la estructura y cuerpo del robot. Por ser un material fácil de maquinarse, considerando que el proceso a utilizar sea corte láser, y contar con la resistencia suficiente para desarrollar el prototipo, sin contar la disponibilidad de este material en el mercado.



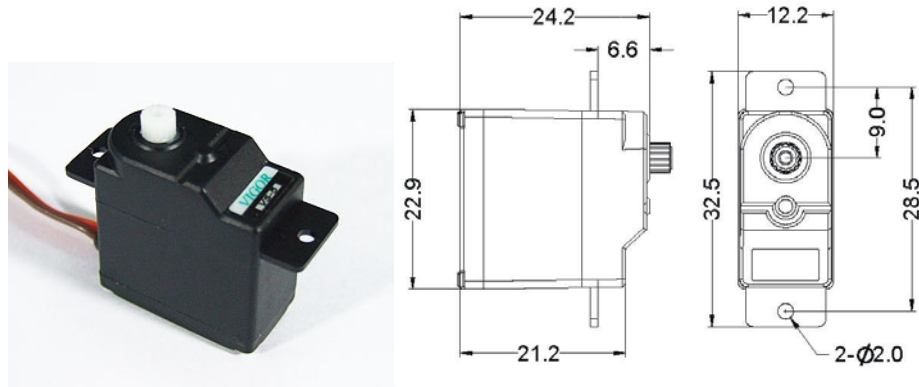


Figura III – 2. Servomotores de Aeromodelismo a utilizar.

Otra razón muy importante para las decisiones que se tomaron, fue el factor económico, ya que, desarrollar un Manipulador Móvil de escala industrial es de costo muy elevado, debido a los sistemas que utiliza. Utilizando motores CD para las juntas del manipulador, aumentaría las dimensiones del mismo, sin contar la etapa de potencia y el control para obtener los pares. Otro punto importante, es que el material con el que están hechos, es mucho más resistente y los procesos de manufactura para su formado son más caros y más complejos. Por lo que se incrementaría el precio final del producto.

III. 1. 2. DISEÑO MODULAR

Por las razones anteriores, se tiene que diseñar un Manipulador Móvil de arquitectura modular. Si la tarea que se planea realizar es diferente a la que plantea este trabajo, simplemente se tendría que cambiar alguno de los subsistemas y se sustituye en el sistema. Esto, con la finalidad de que el robot, pueda ser utilizado en trabajos futuros, que desarrollen miembros del grupo de trabajo, ajenos al autor. Los subsistemas que se mencionan son: la plataforma móvil, el brazo manipulador y el efector final. Los tres, se desarrollan de manera separada en las siguientes secciones y al final de este capítulo, se habla sobre la arquitectura en su totalidad. El software utilizado para realizar el modelo del manipulador móvil fue *Unigraphics NX7*, de *Siemens*. De aquí en adelante, todas las imágenes que muestren el modelo del robot, fueron realizadas en este software.

III. 1. 2. 1. PLATAFORMA MÓVIL

La plataforma móvil se realizó inspirada en la plataforma de *Neobotix*, que se muestra en la Figura II – 7, esto, debido al material disponible para su construcción. Lo primero que se hizo, fue modelar el material disponible (motores,

soportes, llantas), con la finalidad de tener una idea de cómo va a quedar el modelo antes de mandar a maquinar.

Una vez que las medidas de este material se obtuvieron, lo siguiente fue proponer una medida y forma de la plataforma, la geometría se realizó similar a la de la plataforma de *Neobotix* (Figura III – 3). Con el propósito de tener un espacio donde transportar material o algún objeto. La pieza que aparece en la figura es, para futuras referencias, la placa de transporte. El espacio tiene la finalidad de que, la plataforma no solo realice una tarea específica, sino que pueda utilizarse para desarrollar otras tareas. Las dimensiones se propusieron, tomando en cuenta el largo de los motores de CD (locomoción de la plataforma), y las dimensiones de la batería (encargada de proporcionar el voltaje de alimentación del robot), para el ancho de la plataforma. La medida del largo de la plataforma, fue propuesta por el autor de este trabajo. Finalmente, con la forma y dimensiones propuestas, se ubican perforaciones en puntos cercanos a los vértices de la placa de transporte, para realizar el ensamble entre las demás piezas que forman la plataforma, así como darle rigidez a la estructura. Las dimensiones se muestran en la Figura III – 3, el grosor de esta placa es de 6 mm, también se muestra la dimensión de las perforaciones. El diámetro de las perforaciones, es el mismo que en las demás placas, debido a que es donde se van a unir todas las piezas. Por lo cual, no se vuelven a indicar en el resto de la sección.

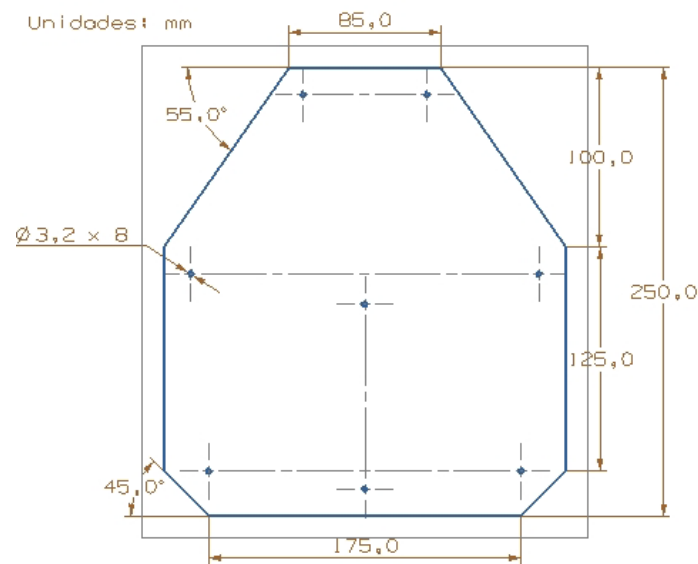


Figura III – 3. Forma y dimensiones de la placa de transporte.

El siguiente paso, fue determinar el tamaño de la placa base, donde los motores de CD se encuentran montados. Para esto, se tomaron las medidas de la placa de transporte y se dividió en dos partes: la parte trasera de la placa de transporte, formando la placa base; y la parte delantera o placa de apoyo, que se

menciona más adelante. Las medidas de la pila se muestran en la Figura III – 4. La pila se apoya sobre su cara de 70 mm, en la placa base. Los soportes de los motores, se encuentran unidos a la placa base, siendo los motores los que impiden el desplazamiento de la pila a lo largo del eje de transmisión, como se aprecia en la Figura III – 5. Para impedir el desplazamiento a lo largo del eje de movimiento de la plataforma, existe una perforación por el cual se unirá una pieza, que sirve como poste para detener la pila, de tal manera que ésta no se salga del compartimiento. Además este elemento funge como elemento de ensamble.

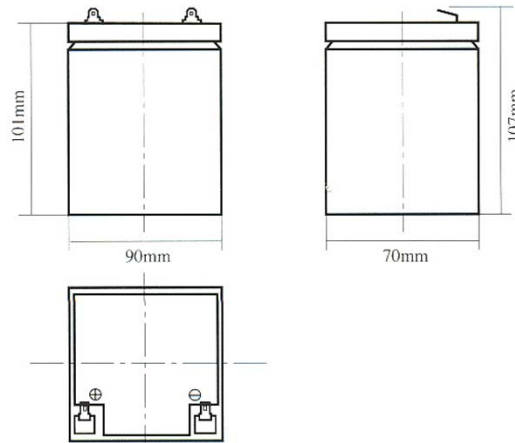


Figura III – 4. Medidas de la pila utilizada.

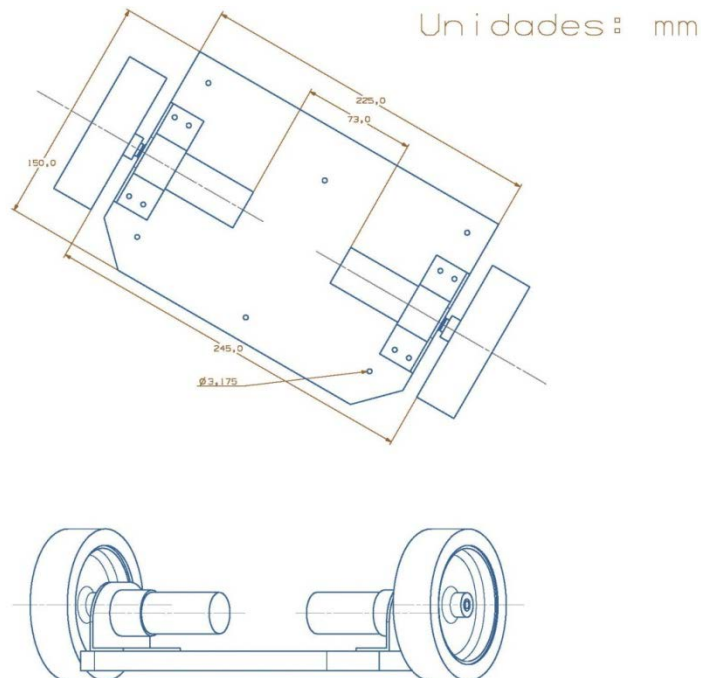


Figura III – 5. Modelo en Nx7 de la placa base y sus componentes.

En el lado opuesto, de donde se ubica el poste que detiene la pila, se encuentra el borde de la placa base. En este lado, no hay nada que impida el

desplazamiento de la pila. Para que ésta, no se salga del compartimiento, se coloca otra placa, que se encuentra entre la placa de transporte y la placa base. Esta placa es: la placa de apoyo (Figura III – 6a), entre sus funciones se encuentra: impedir el desplazamiento de la pila, espacio para colocar sistemas auxiliares, y la más importante, soporte para la colocación de la rueda de apoyo. Debido a que la configuración de la plataforma es diferencial (se trata con mayor detalle, en el capítulo 4), se tienen dos ruedas motrices y se necesitan ruedas pasivas que den soporte a esta configuración, es decir, se necesita una rueda no actuada que dé soporte a la plataforma. De lo contrario, ésta no estaría balanceada y se encontraría oscilando sobre el eje de tracción de los motores. Por esto, se necesita incluir la rueda en la parte delantera de la plataforma. A la rueda mencionada se le conoce como: rueda loca. Si se requiere mayor estabilidad, otra rueda en la parte trasera sería necesaria, sin embargo, mientras el cambio de velocidad no sea repentino, es suficiente una sola rueda. En la Figura III – 6b, se muestra el modelo de la rueda loca que se utilizó y algunas de sus dimensiones. Al igual que en las otras placas, se encuentra cierto número de perforaciones, para su ensamble.

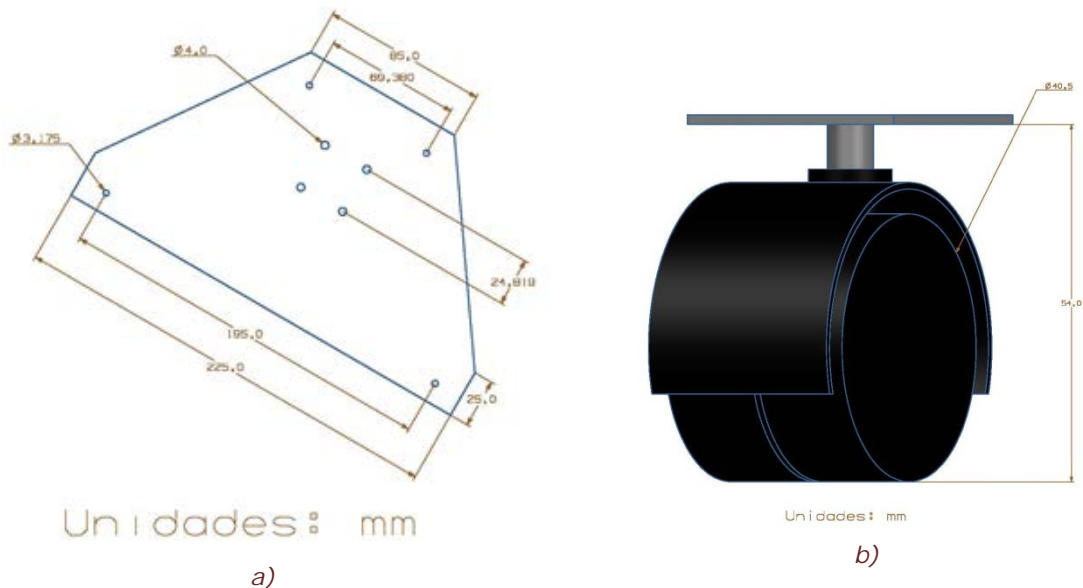


Figura III – 6. Placa de apoyo (a) y modelo de la rueda loca utilizada (b).

Existe una última placa, la cual, es exactamente igual a la placa de transporte y se encuentra debajo de ésta. La diferencia es que hay un espacio entre ellas, para cualquier dispositivo necesario controlador de la plataforma. Por ejemplo, circuitos, tarjeta controladora, etapa de potencia, entre otros. Finalmente, teniendo todas las placas, se ensamblan. Las piezas se ensamblan por medio de postes, formados por un vástago roscado y un tubo. Además de ser espaciadores, fungen como estructura de la plataforma. Las dimensiones de los postes están calculadas a manera que, la pila esté contenida en el compartimiento, sin

problemas de espacio, dejándole un claro de no más de 5 cm con la placa de transporte. Las perforaciones de las placas, están hechas para permitir el paso del vástago roscado, que forman los postes. Mediante arandelas y tuercas se unen las placas, a modo que se impida el desensamble. El resultado del proceso anterior, se muestra en la Figura III – 7a, en la que se muestran las cuatro placas ensambladas, así como los elementos de locomoción y la rueda loca. Sin embargo, esta plataforma tiene que ser utilizada para formar un manipulador móvil, por lo que, un brazo manipulador debe ir montado en la plataforma. Así que es necesario quitarle una placa de transporte para disminuir la distancia al piso. Mientras mayor sea esta distancia, mayor trabajo le costará al manipulador llegar al nivel del piso, donde está ubicado el objeto a recoger. Otro factor es que, el área disponible de la placa de apoyo, es suficiente para los componentes que se encargarán de los movimientos de la plataforma móvil y del brazo manipulador, para la aplicación aquí presentada. Por lo anterior, el modelo de la plataforma móvil a maquinar, es el que se muestra en la Figura III – 7b.

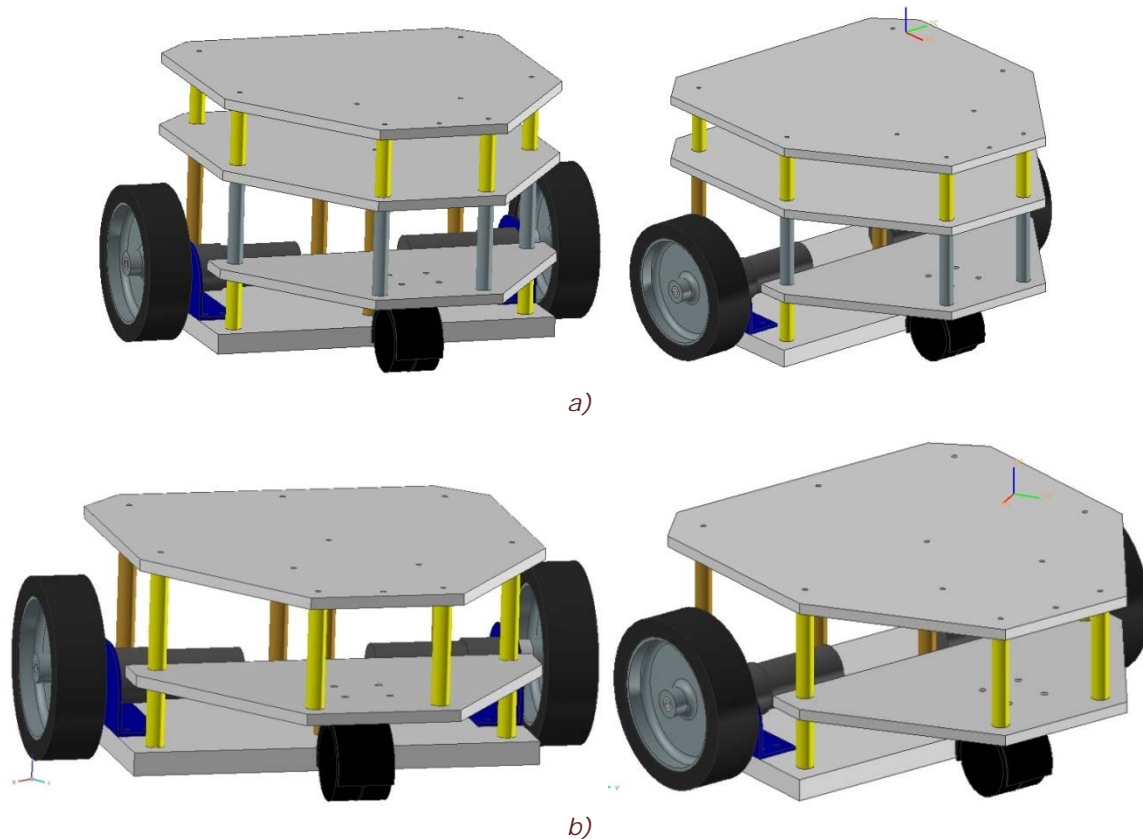


Figura III – 7. Dos diferentes versiones de la plataforma móvil: a) Con dos placas de transporte, b) Con una placa de transporte.

III. 1. 2. 2. BRAZO MANIPULADOR

El brazo manipulador, tiene que cumplir con la característica de ser antropomórfico. Para un manipulador espacial los primeros tres GDL (base, hombro, codo), son los encargados de ubicar el efector final en cualquier posición del espacio. Los últimos grados de libertad, forman la muñeca, encargada de orientar el EE para la tarea a realizar. Para el manipulador a desarrollar, se tienen dos juntas, las cuales, cumplen con el objetivo de orientar al EE: inclinación de la muñeca y giro sobre su propio eje, con en el propósito de formar una estructura sencilla.

Como se indicó anteriormente, se utilizan servomotores de aeromodelismo, para proporcionar movimiento al manipulador. Se modelaron estos dispositivos en CAD, con el fin de encontrar las medidas adecuadas para su instalación y determinar la forma en que se unirán a los eslabones. Además de que proporciona una idea previa de cómo el manipulador quedará, al momento de ser manufacturado.

Antes de detallar el manipulador definitivo, se muestran los conceptos previos al definitivo. El primer concepto, fue desarrollar al manipulador con una arquitectura tipo PUMA (Figura III – 8). Se escogió en un principio, por ser una arquitectura estudiada en la asignatura de robótica. Su cinemática directa e inversa ya estaba estudiada y se contaba con más conocimiento sobre ésta. El principal inconveniente fue, que con esta arquitectura uno escoge si es hombro derecho u hombro izquierdo, quiere decir, que existe una distancia de desplazamiento entre la primera junta y la segunda. Este desplazamiento es contraproducente, si se escogen servomotores de aeromodelismo, debido a que estos no están hechos para soportar pares en su eje de salida. Por lo que, es de gran importancia, evitar o reducir los desplazamiento entre los ejes de las juntas.

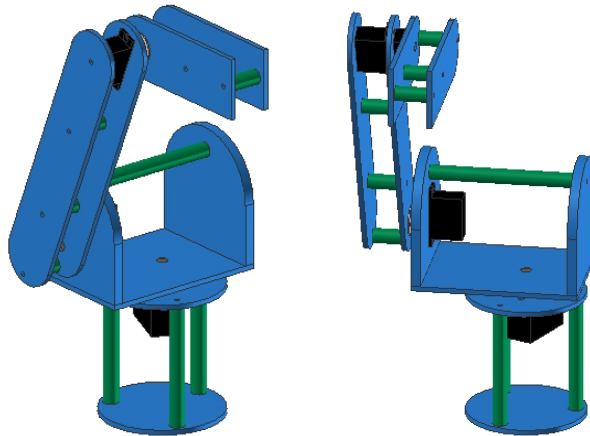


Figura III – 8. Modelo de arquitectura tipo PUMA.

El segundo concepto, fue una arquitectura similar a KUKA *youBot* de cinco GDL. En esta arquitectura se cuenta que, el eje de la primera junta coincide con el eje de la segunda, es decir, no hay hombro derecho ni izquierdo, sino, se encuentra en medio. Esta configuración se vuelve la definitiva, con la excepción de un detalle; los servomotores de tamaño estándar, entregan un par relativamente alto, para esta aplicación; a pesar de eso, suele no ser suficiente. En primera instancia, se propusieron las dimensiones para los eslabones y se desarrollaron los modelos en CAD. Tomando en cuenta las dimensiones propuestas, se realiza un diagrama de momentos (Figura III – 9), en donde, al modelo se le asignan las propiedades del material (acrílico) y, con ayuda del software, se calcula la masa de cada eslabón. Aprovechando su geometría simétrica, se considera que el centro de masa de cada eslabón, se encuentra en el punto medio. Por las hojas de especificaciones de los servomotores, se ubica la masa de estos y su par máximo. Todas estas medidas se muestran en la Figura III – 9.

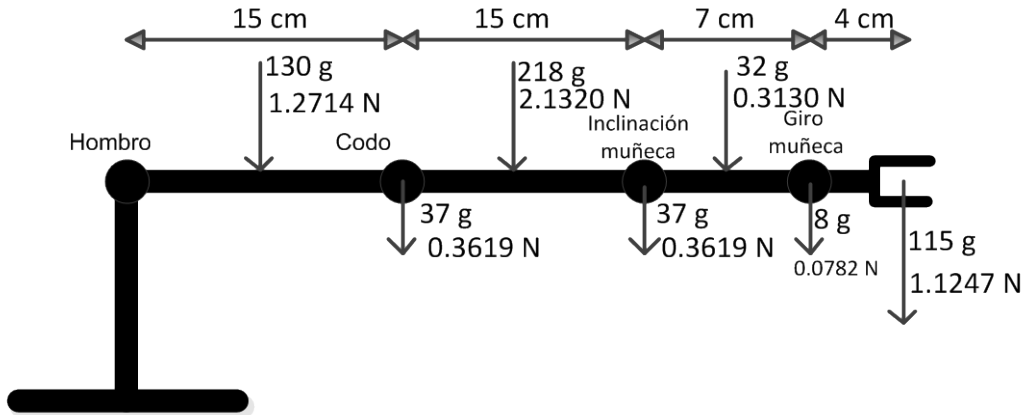


Figura III – 9. Diagrama de momentos para las juntas del manipulador.

Se realiza el diagrama en la posición donde el brazo de palanca es el mayor, para cada servomotor, es decir, el caso más crítico para cada junta, sin contar la junta de la base, además se considera que cargará un objeto de aproximadamente 20 gramos. Teniendo en consideración todos los datos anteriores, se calcula el par para las dos juntas de mayor importancia: el hombro y el codo. Éstas son las que se encuentran sujetas a mayor carga, obteniendo lo siguiente:

$$M_{\text{máximo}_{\text{hombro}}} = 1.3328 \text{ Nm} = 13.6280 \text{ kg}_f \cdot \text{cm}$$

$$M_{\text{máximo}_{\text{codo}}} = 0.5592 \text{ Nm} = 5.7180 \text{ kg}_f \cdot \text{cm}$$

Se sabe que un solo servomotor de tamaño estándar no entrega un par tan alto, como el calculado para el hombro. Por lo tanto, se dispone a adecuar el manipulador de forma que, para la junta del hombro, se tenga dos servomotores. Los dos servos, en conjunto, deben entregar el par suficiente, para desempeñar la tarea. En el caso del codo es más difícil colocar dos servomotores, ya que, el

hombro se encuentra siempre soportado sobre la junta base independientemente de la posición angular que tome la base. Sin embargo, el hombro y codo se encuentran desplazados por la distancia entre sus juntas. El hombro, en todo momento, transporta a los eslabones posteriores. Agregar otro servomotor, aumentaría la masa de los eslabones y habría que calcular de nuevo los pares máximos de las juntas. Por lo que desde un principio, se debe considerar un servomotor estándar, de un par muy alto o un servomotor estándar, de par no tan alto y que se encuentre al máximo de su capacidad. Por esta razón, se modificó el modelo original dando lugar al definitivo, que se describe en los siguientes párrafos.

Para empezar a desarrollar el modelo en CAD del brazo manipulador definitivo, primero hay que definir dos conceptos que se aplican, para el ensamble de las juntas y para la sujeción de los servomotores al manipulador.

El método que se selecciona, para ensamblar las piezas que formarán al manipulador es, por caja y espiga (Figura III – 10a). Este es un método de ensamble muy utilizado en la carpintería, para la fabricación de marcos de muebles, marcos de puertas, ventanas, entre otros. El ensamble es simple: la espiga se inserta en la caja y se mantiene unidas, dando la apariencia de ser una sola pieza. Si se realiza un ajuste entre estos elementos, la espiga entraría a presión y el ensamble se consideraría permanente, entonces el manipulador no tendría por qué desarmarse. Esta manera de ensamblar las piezas, se realiza por su facilidad de ensamble; no necesita otro elemento (tornillos, clavos, etc.) para unirlos y distribuye los esfuerzos, en toda la estructura, sin necesidad de elementos de unión. Finalmente, para agregar más resistencia al ensamble, colocar un adhesivo entre las piezas involucradas, ayuda a lograrlo.

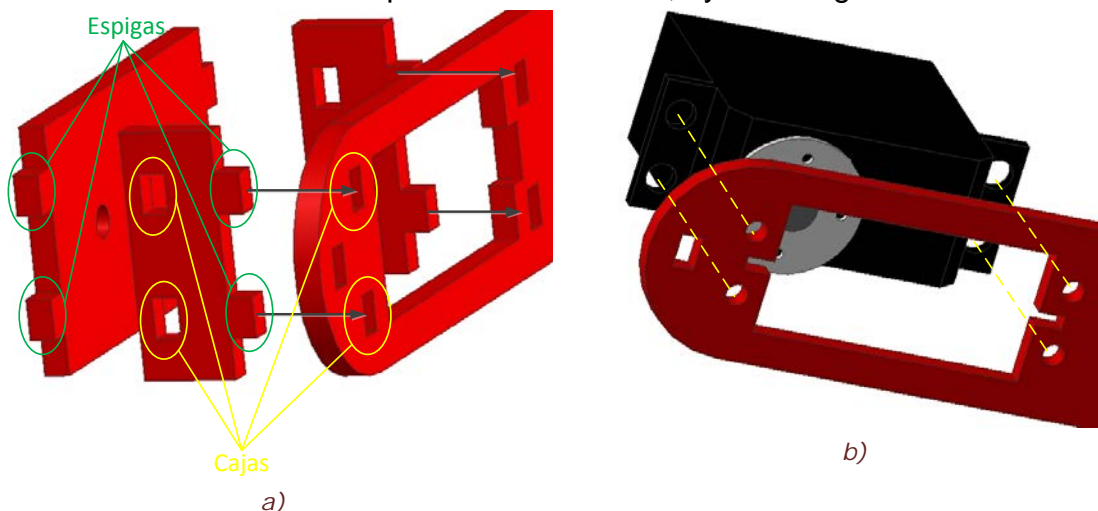


Figura III – 10. Consideraciones previas: a) Ensamble espiga y caja, b) Sujeción de servomotores.

En la Figura III – 10b, se muestra la forma de unir los servomotores a los eslabones. Los servomotores vienen con perforaciones de fábrica, que sirven para colocar tornillos y tuercas y así, unirlos a la estructura que los contiene. Esta característica se aprovecha: haciendo una ranura en los eslabones donde se monta. Los orificios mencionados, que coinciden con los del servomotor, lo sujetan a la pieza que se necesite, mediante el uso de tornillos, arandelas de presión y tuercas. El manipulador cuenta con dos tamaños diferentes de servomotor, para cada uno, se realiza el mismo sistema de sujeción, pero, con diferentes medidas.

La primera junta que se va a formar es la base, la más sencilla, pues es la que menos piezas tiene y la que menos interfiere con las demás juntas. Ésta cuenta con tres piezas principales: pieza de movimiento, pieza de sujeción de servomotor y pieza base (Figura III – 11). La pieza de sujeción, es en la que el servomotor se encuentra sujeto, exponiendo el vástago (eje de salida del movimiento) del servomotor. El vástago, se acopla a su aditamento para transmitir movimiento. El aditamento se une a la pieza de movimiento, por lo tanto, ésta es la pieza que conecta: la junta base con la junta del hombro. Entre estas piezas, hay movimiento relativo. En la pieza de movimiento, se encuentran las cajas donde entran las espigas que forman la siguiente junta (recordando que la junta del hombro está compuesta de dos servomotores). En la pieza de sujeción hay seis orificios, en los cuales se colocan ruedas locas, éstas funcionan como cojinete para la pieza de movimiento, reduciendo fricción y aportando más estabilidad al movimiento entre las piezas. Finalmente, la pieza base, como el nombre lo indica, es la pieza que forma la base del manipulador, es decir, el origen del sistema del manipulador. Hay una distancia entre las últimas dos piezas, principalmente definida, por la altura del servomotor.

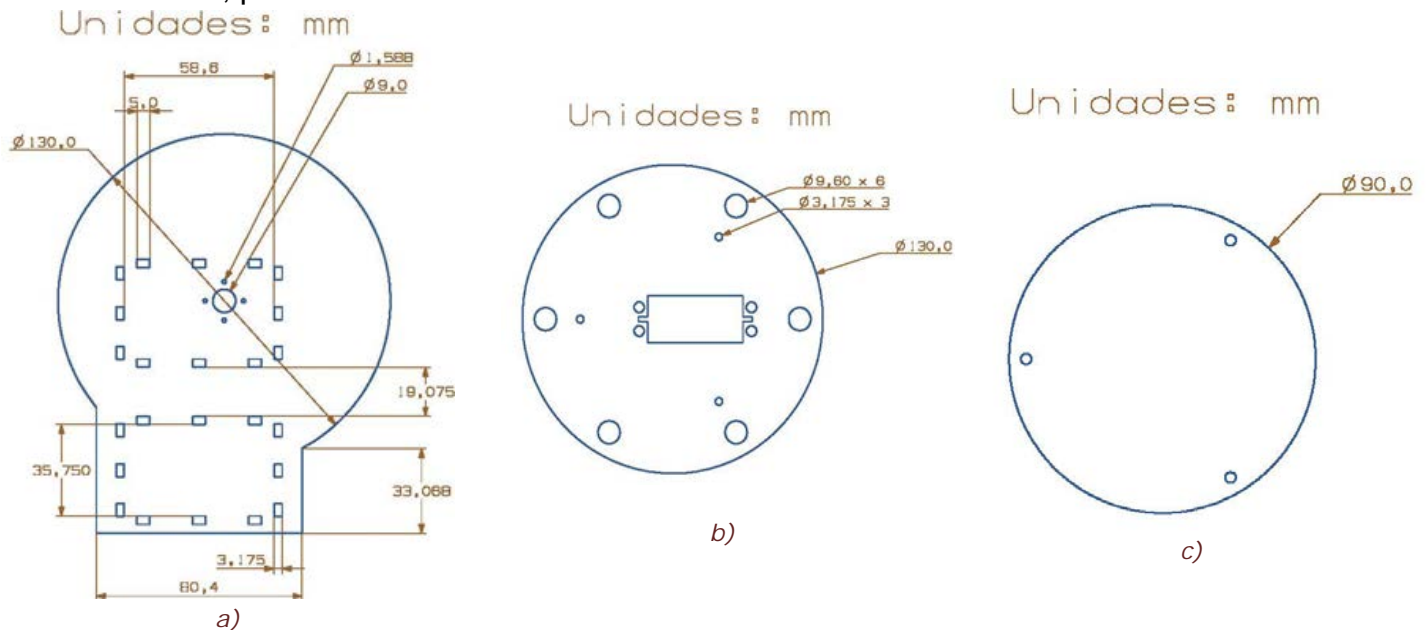


Figura III – 11. Piezas que forman la base del manipulador: a) pza. de movimiento, b) pza. de sujeción, c) pza. base.

La siguiente junta, es la del hombro. Está formado por dos servomotores, que transmiten el movimiento al eslabón dos. Los dos servomotores se conectan al eslabón uno, uniendo las espigas (encontradas en la parte inferior del marco que contiene a los servomotores), con las cajas encontradas en la pieza de movimiento del eslabón uno. En la Figura III – 12a, se muestran las piezas que forman la conexión del eslabón uno con el dos. Al igual que con la junta base, los servomotores se unen mediante su aditamento al eslabón dos. Estas piezas son, las que se mueven respecto a la junta del hombro, y se muestran en la Figura III – 12b. El primer servomotor, se encuentra dentro de las piezas que forman al segundo eslabón. El segundo está fuera del eslabón, de tal modo que, ambos realizan el mismo movimiento ya que sus ejes son colineales.

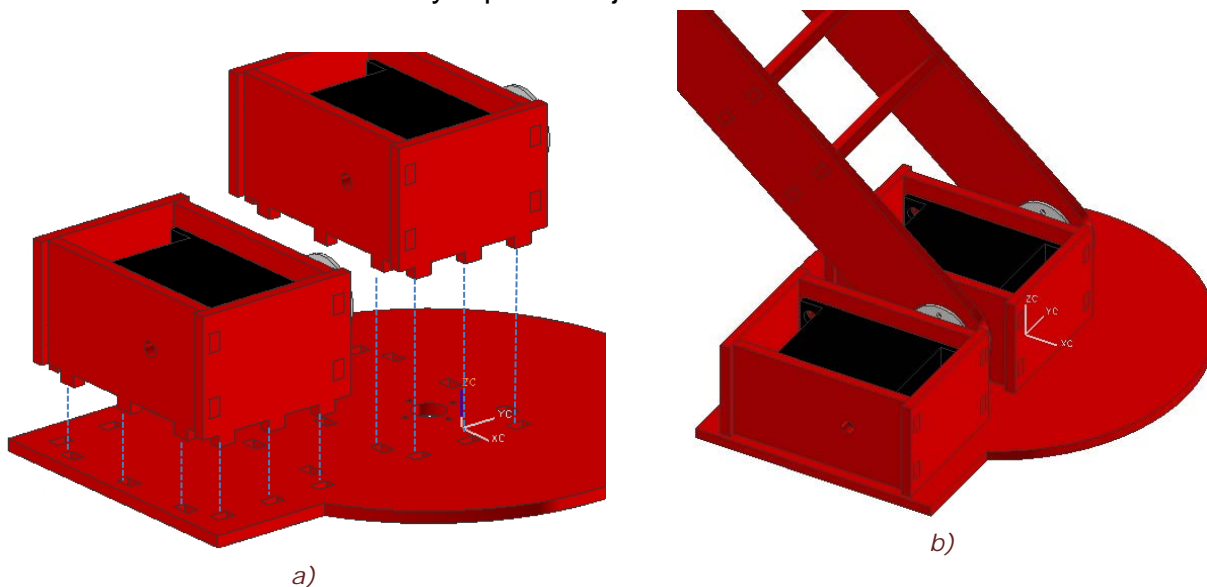


Figura III – 12. Piezas ensambladas que forman junta del hombro y el segundo eslabón.

La distancia entre las juntas del hombro y codo, es decir, la longitud del eslabón dos, es propuesta del autor y se muestra en la Figura III – 9. Es de particular cuidado esta junta, pues consiste de dos servomotores. Por lo tanto, estos, deben moverse a una misma posición, así que deben estar alineados en todo momento.

El eslabón dos está formado: por dos piezas planas paralelas, las cuales se conectan a la junta del hombro. Debido a que los servomotores sólo cuentan con un vástago para el eje de salida. Una de las dos piezas del eslabón dos, está unida al aditamento del servomotor y la otra, se encuentra sin apoyo para el movimiento del eslabón tres. Por este motivo, es necesario realizar un apoyo, en la pieza que no sujeta al servomotor. La solución propuesta se muestra en la Figura III – 13, ésta consiste de un marco, que se une a la pieza de apoyo del eslabón tres. El marco contiene un orificio colineal al eje de salida de movimiento

del servo, para la junta del codo. En este orificio pasará el eje, que permite el movimiento relativo entre ambas piezas. En la pieza del eslabón dos, se colocan dos discos, uno en cada lado de la pieza y se unen entre sí. Para reducir el área de fricción entre las piezas de movimiento.

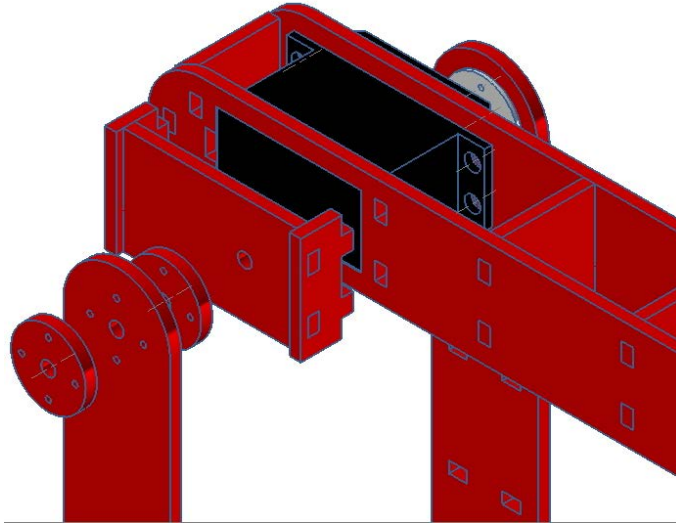


Figura III – 13. Conexión entre eslabón dos y eslabón tres

La junta tres, consiste de un sólo servomotor. A diferencia de las demás juntas, el servomotor se mueve junto con el eslabón tres. Esto quiere decir, que el acoplamiento unido al eslabón dos, se mantiene fijo, y el servomotor se mueve junto con el eslabón tres. Por lo que el servomotor, que conforma a la junta tres, carga los eslabones posteriores y a sí mismo. El eslabón tres, al igual que el dos, está formado por dos piezas paralelas. Una de las piezas que lo forman, es la pieza de sujeción. Esta pieza, está encargada de sujetar dos servomotores: codo e inclinación de muñeca. La otra, es la pieza de apoyo, en donde se ubican dos ranuras para que los servomotores entren en ésta, proporcionando apoyo a ambos. En ambas piezas, se encuentran las cajas, que permiten entrar a las espigas de cuatro piezas que forman al eslabón tres. El eslabón, la junta del codo y la junta de inclinación de muñeca se encuentran en la Figura III – 14. La distancia entre las juntas del codo y la inclinación de muñeca se encuentran en la Figura III – 9. El eslabón tres, en sentido estricto, contiene a la junta de inclinación de muñeca. En esta pieza se encuentra, el servomotor y el mismo sistema de apoyo de movimiento que conecta al eslabón dos con el tres. Por lo anterior, el vástago de salida de la junta, inclinación de muñeca, se une al eslabón cuatro. Al igual que la junta del hombro, el servomotor mueve al eslabón siguiente.

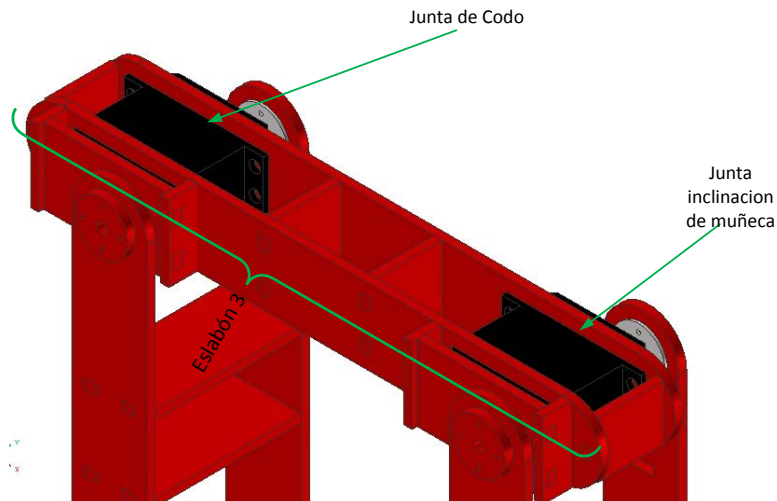
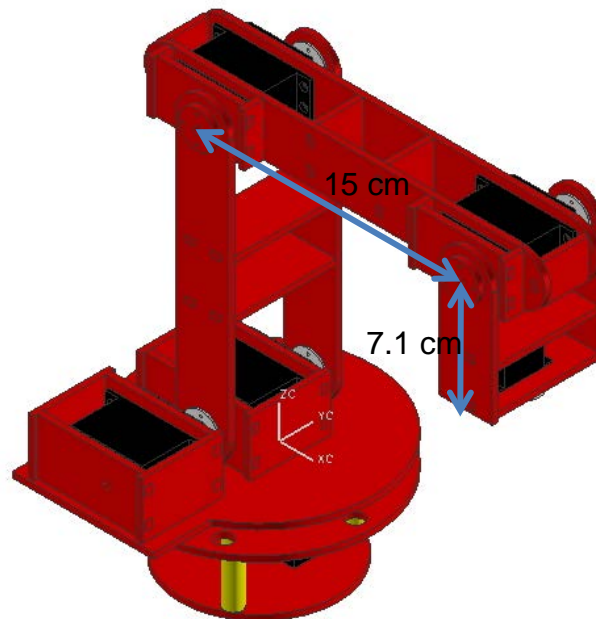


Figura III – 14. Piezas que forman al eslabón tres y los componentes que contiene.

Finalmente, el eslabón cuatro consta de dos piezas paralelas: las cajas para el ensamble, y dos piezas perpendiculares a éstas: las espigas para su unión. En una de las piezas perpendiculares, está sujeta la última junta de orientación de muñeca, la que conecta con el siguiente módulo: efector final. La distancia entre las juntas de orientación, se muestra en la Figura III – 9. En la misma figura, se muestra la distancia de la última junta, al punto donde el objeto es manipulado.

Como resultado de todo el proceso descrito en esta sección, se muestra en la Figura III – 15, el modelo en CAD del manipulador a utilizar.



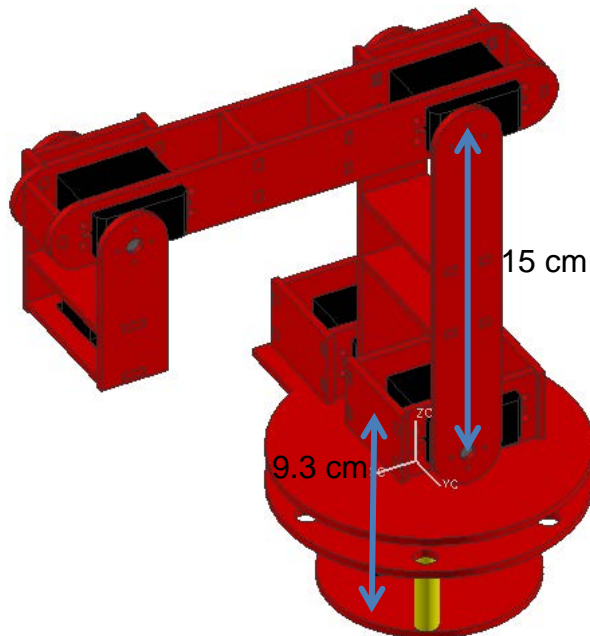


Figura III – 15. Modelo en NX7 del brazo manipulador utilizado.

III. 1. 2. 3. EFECTOR FINAL.

El efector final, para el presente trabajo, consiste de un *gripper* con dos dedos paralelos. El objeto a manipular, no es de geometría compleja, por lo tanto, el *gripper* utilizado cumple con el objetivo. El método para ensamblar las piezas es el mismo que se utiliza en la sección anterior: espiga y caja. Es importante señalar, que el modelo está inspirado en uno ya existente.

En este modelo, el primer punto a abordar es la ubicación del servomotor, éste debe estar sujeto al mismo *gripper*, pues no tiene contacto con otro eslabón. Entonces, las piezas que sujetan al servomotor, cumplen con otras funciones que se detallan más adelante. El servomotor, se encuentra confinado al cuerpo del *gripper*, éste está formado por dos piezas que sujetan al servo de manera transversal. En estas dos piezas, se ubican las espigas, para ensamblar las paredes del cuerpo que forma el *gripper*, que a su vez, se hicieron para darle rigidez al cuerpo, así como la pared trasera, sirve para unirse a la última junta del manipulador. La pared delantera de la estructura, junto a las dos piezas unidas a las piezas de sujeción, funcionan como paredes para la corredera. La corredera, está encargada de desplazar los dedos del *gripper*, como se aprecia en la Figura III – 16, las piezas que conforman esta primera parte del *gripper*. En este modelo, se aprecia de mejor forma, lo útil que es el método de ensamble por caja y espiga.

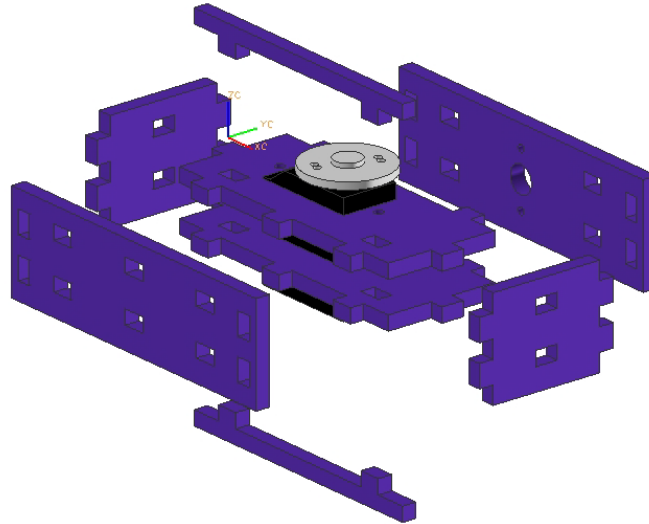


Figura III – 16. Piezas que forman la estructura principal del gripper.

Para realizar la operación de abrir y cerrar *gripper* se utiliza un mecanismo de biela – manivela. La manivela, es la rueda que se acopla al eje de salida del servomotor, que gira de acuerdo al cierre o apertura del *gripper*. Esta manivela cuenta con dos bielas a 180°, cada una se encarga de describir el movimiento para cada dedo. El mecanismo está representado en la Figura III – 17a. Las bielas, permiten realizar un movimiento rectilíneo a través de los dedos del *gripper*, y asimismo, se unen a ellas mediante pernos. En estos dedos (Figura III – 17b), se encuentran las correderas, que se desplazan dentro de la zona mencionada en el párrafo anterior. Son dos correderas, una en la parte superior y la otra en la parte inferior del cuerpo del gripper. Se realizaron dos correderas, pues, el dedo está formado por las piezas acopladas a las bielas, y en estas piezas se encuentran los vástagos de las correderas. A lo largo de la pared delantera, se encuentran las piezas que permiten sujetar al objeto que se requiere manipular. Estas placas están paralelas entres sí (de ahí, que el *gripper* sea de dos dedos paralelos), de forma que, al momento de cerrarse, el área de contacto sea amplia, y por ende, que el objeto no tienda a caerse o deformarse. Mediante el mecanismo propuesto y las piezas utilizadas para el *gripper*, se logra el movimiento que se muestra en la Tabla III – 1.

Distancia entre dedos	Angulo de servomotor
1.5 cm	0°
3.5 cm	55°

Tabla III – 1. Tabla de correspondencia para abrir y cerrar gripper.

El resultado de este proceso se muestra en la Figura III – 18, con todas las piezas ensambladas. Un componente que no se muestra en la figura, es el perno

que fue realizado con una varilla de acrílico y que sirve para unir las bielas y los dedos.

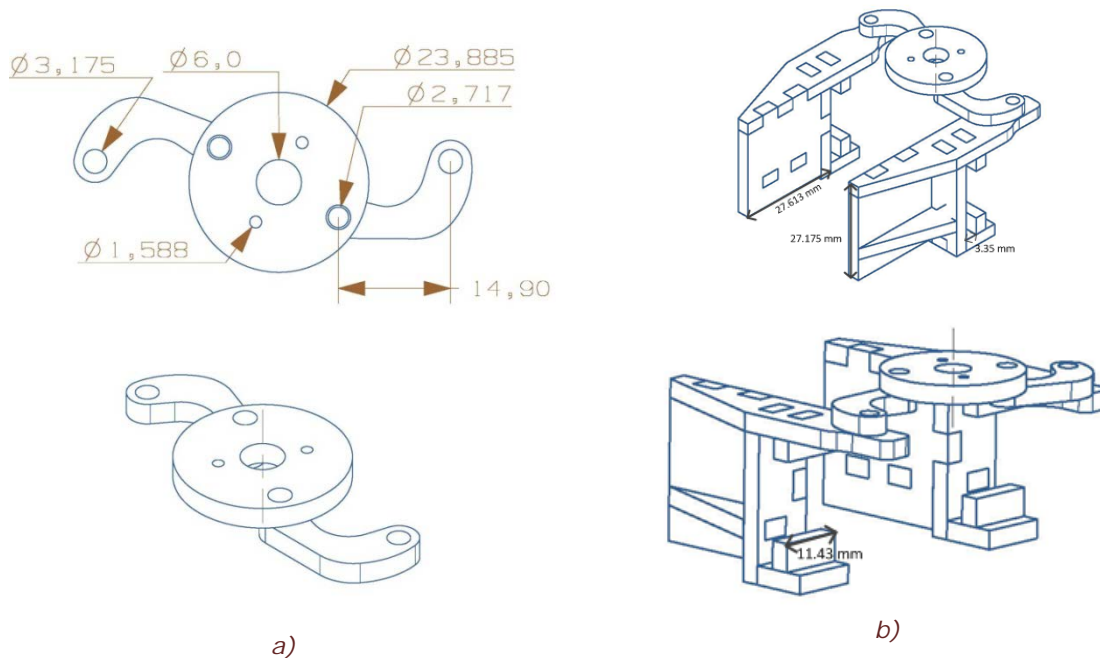


Figura III – 17. Mecanismo biela manivela y corredera encargado de abrir y cerrar gripper.

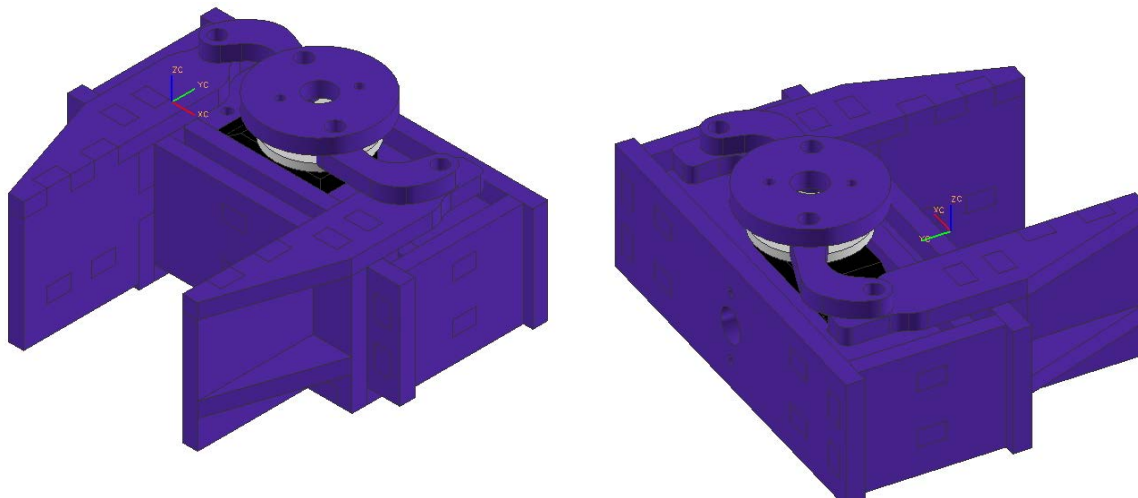


Figura III – 18. Modelo en NX7 del efector final utilizado.

III. 2. ARQUITECTURA.

La arquitectura, es el acomodo adecuado de elementos físicos de un producto, para llevar a cabo sus funciones requeridas. Está constituida por la relación entre los componentes del producto y las funciones a realizar.

Existen principalmente dos estilos de arquitecturas la integral y la modular. En la arquitectura modular, cada módulo implementa una o pocas funciones y la

interacción modular está bien definida. Una arquitectura integral, reúne las funciones en uno o pocos módulos.

El manipulador móvil que se presenta en esta tesis, consta de una arquitectura modular. Una ventaja de que la arquitectura del robot sea modular, es que puede estar actualizándose con mayor facilidad en el transcurso del tiempo. Otra ventaja es que, acorde a las necesidades del trabajo que esté llevando a cabo, los módulos pueden ser intercambiables, como consecuencia el prototipo tendrá mayor tiempo de vida y los módulos pueden ser utilizados en diferentes actividades.

Cada módulo del Manipulador Móvil, posee una interfaz para la conexión entre módulos. A través de las interfaces, los módulos se pueden conectar entre sí para que cumplan con la característica correspondiente. Por ejemplo, si se realiza otro brazo manipulador, se puede acoplar a la plataforma móvil, siempre y cuando exista la interfaz correspondiente entre ellos. A continuación, se mostrarán las diferentes interfaces para cada módulo.

Para la plataforma móvil, la interfaz está en la placa de transporte. En esta placa se encuentran tres perforaciones, a través de ellas se conecta al siguiente módulo. En la Figura III – 19a, se muestran las dimensiones de las perforaciones, que forman un arreglo de tres orificios, que describen un triángulo equilátero. En el inciso b) de la figura, se muestra la pieza base del eslabón uno, en la que se aprecia la misma disposición de orificios; estos conectan la plataforma móvil, con el brazo manipulador. En el brazo se encuentra la interfaz para conectar al siguiente modulo, ésta se ubica en el aditamento del servomotor de la última junta. En la interfaz del manipulador, también, cabe la posibilidad de cambiar el aditamento y así cambiar las dimensiones de la interfaz. Finalmente, en el inciso c) se muestra la interfaz correspondiente a la última junta del manipulador, aquí se aprecian las dimensiones la interfaz. Los tres módulos se unen mediante tornillos y tuercas.

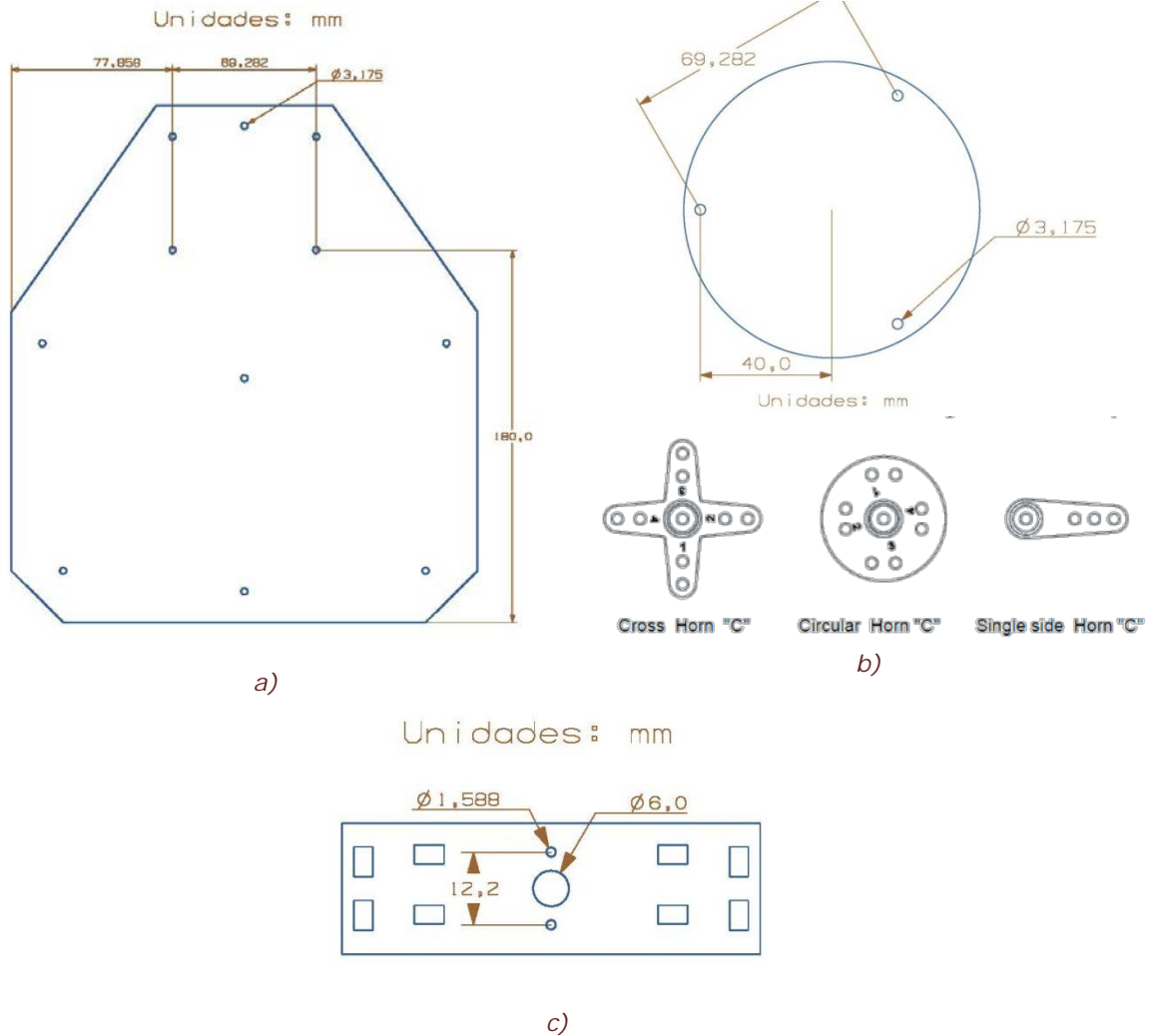


Figura III – 19. Interfaces entre módulos: a) plataforma móvil, b) brazo manipulador y c) efector final.

Con los tres módulos terminados y habiendo identificado las interfaces que los unen entre sí, se arman todas las piezas, dando lugar al completo ensamble del sistema. En la Figura III – 20, se muestra el resultado del diseño del Manipulador Móvil.

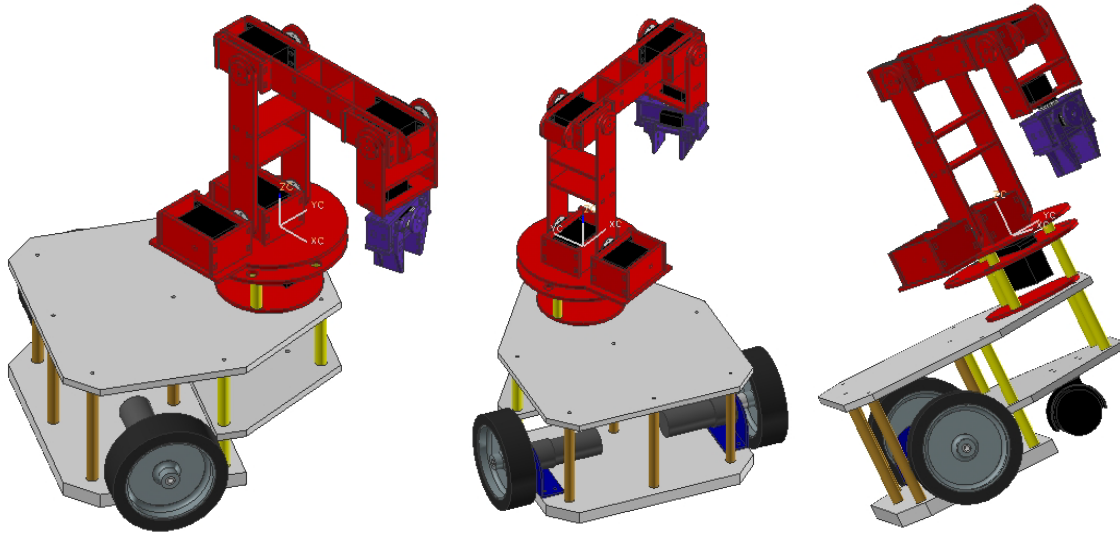


Figura III – 20. Manipulador Móvil diseñado en NX7.

III. 2. 1. LISTA DE COMPONENTES, MATERIALES Y PIEZAS.

Con todo lo que se ha descrito a lo largo de este capítulo, se tiene el modelo completo en CAD, para el Manipulador Móvil. La Tabla III – 2, detalla una lista de todos los componentes, materiales y piezas utilizados para realizar el Manipulador Móvil.

Cantidad	Componente	Descripción
Plataforma móvil		
2	Piezas de acrílico cristal de 6mm.	Placa de transporte y placa de apoyo.
1	Pieza de acrílico cristal de 8mm.	Placa base.
1	Rueda loca de nylon.	Rueda de apoyo.
1	Tarjeta MD25	Tarjeta controladora para dos motores EMG30.
2	Motores EMG30.	Motores de DC que proporcionan movimiento a la plataforma.
2	Soportes para motores.	Soportes para motores EMG30
2	Ruedas de 10 cm de diámetro.	Ruedas motrices de la plataforma.
1	Batería ácido-plomo 12v 4AH.	Batería marca Steren para alimentar al robot.
4	Espaciadores de 10cm	Encargados de mantener separadas y unidas las placas de transporte y la placa base.
4	Espaciadores de 5.8cm	Encargados de mantener separadas y unidas las placas de transporte y la placa de apoyo.
2	Espaciadores de 3.6cm	Encargados de mantener separadas y unidas las placas base y de apoyo.
4	Tornillos de 5/32" x 1/2"	
4	Arandelas para tornillos de 5/32" x 1/2"	Para unir la rueda loca a la placa de apoyo.
4	Tuercas para tornillos de 5/32" x 1/2"	
8	Tornillos de 1/8" x 1/2"	Para unir los soportes de los motores a la placa base.
8	Arandelas de presión para tornillos de	

	1/8"	
8	Tuercas para tornillos de 1/8"	
20	Arandelas de presión para tornillos de 1/8"	Para unir los espaciadores a las placas correspondientes, tres de estos juegos corresponden a la interfaz entre plataforma y manipulador.
20	Tuercas para tornillos de 1/8"	

Brazo manipulador

3	Piezas de acrílico rojo de 3mm.	Primer eslabón del brazo manipulador.
15	Piezas de acrílico rojo de 3mm.	Segundo eslabón del brazo manipulador.
12	Piezas de acrílico rojo de 3mm.	Tercer eslabón del brazo manipulador.
6	Piezas de acrílico rojo de 3mm.	Cuarto eslabón del brazo manipulador.
5	Servomotores tamaño estándar.	Marca Vigor, modelo VS – 2A, que forman las primeras cuatro juntas del manipulador, recordando que la junta del hombro consiste de dos servomotores.
1	Servomotor tamaño mini.	Marca Vigor, modelo VS – 5, última junta del manipulador.
20	Tornillos de 1/8" x 1/2"	
20	Arandelas de presión para tornillos de 1/8"	Para fijar los servomotores a cada eslabón, cuatro para cada servomotor de tamaño estándar.
20	Tuercas para tornillos de 1/8"	
2	Tornillos de 2mm x 8mm	Para fijar el servomotor de tamaño mini al cuarto eslabón.
2	Tuercas para tornillo de 2mm	
32	Tornillos de 2mm x 8mm	Para fijar los aditamentos de cada servomotor encargados de transmitir movimiento a los eslabones correspondiente. Ocho de estos tornillos se utilizan para unir los eslabones que con cuentan con aditamento al marco de apoyo.
2	Postes de 12mm	Funcionan como ejes de movimiento entre los eslabones 2 – 3, y los eslabones 3 – 4.
6	Ruedas locas de plástico.	Ruedas tipo roll-on que funcionan como cojinete para el movimiento relativo entre el eslabón uno y dos.
3	Espaciadores de 4cm	Encargados de separar y unir las piezas base y de sujeción del eslabón uno.
3	Espaciadores de 1.2cm	Encontrados entre la pieza de sujeción y la pieza de movimiento, en el extremo que tiene contacto con la pieza de movimiento se encuentra un balín que reduce fricción entre las piezas, se encarga de separar y unir a las piezas mencionadas.

Efactor final

8	Piezas de acrílico azul de 3mm.	Piezas que forman el cuerpo del <i>gripper</i> .
3	Piezas de acrílico azul de 3mm.	Piezas que forman a la manivela y dos bielas.
14	Piezas de acrílico azul de 3mm.	Piezas que forman a los dedos del <i>gripper</i> , siete piezas por cada dedo.
1	Servomotor tamaño mini.	Marca Vigor, modelo VS – 5, proporciona el movimiento para abrir y cerrar <i>gripper</i> .

4	Pernos de 1/8"	Pernos realizados con varilla de acrílico encargados de unir a la manivela con sus bielas, dos pernos por cada biela.
4	Tornillos de 2mm x 10mm	Dos de ellos utilizados para unir el aditamento del servomotor a la manivela y dos que forman el interfaz manipulador-efector final.

Tabla III – 2. Lista de componentes, materiales y piezas que conforman al Manipulador Móvil diseñado.

III. 3. CONSIDERACIONES PARA MANUFACTURA.

El material seleccionado para realizar el modelo físico, es acrílico, como se mencionó al inicio de este capítulo. Las piezas se elaboran planas, con el propósito de que el método de manufactura fuera sencillo y accesible. El principal motivo de que las piezas sean planas, es que solamente requiere un proceso de corte, para realizar las piezas que conforman al manipulador móvil. De acuerdo a esto, se mandaron fabricar las piezas mediante corte láser. La empresa encargada de realizarlo, indicó que las piezas deben de tener una tolerancia de ± 0.1 mm. Por ello, se modificaron las dimensiones de las piezas que componen al Manipulador Móvil. Se realizaron los planos en la escala que lo solicitaron (1:1) y se enviaron para su manufactura. Una vez que las piezas fueron manufacturadas y entregadas, se ensamblaron. Se unieron todos los componentes que lo forman, teniendo como resultado, el prototipo físico para realizar la tarea a la que fue diseñado. En la Figura III – 21, se muestra el prototipo físico armado y listo para operar.

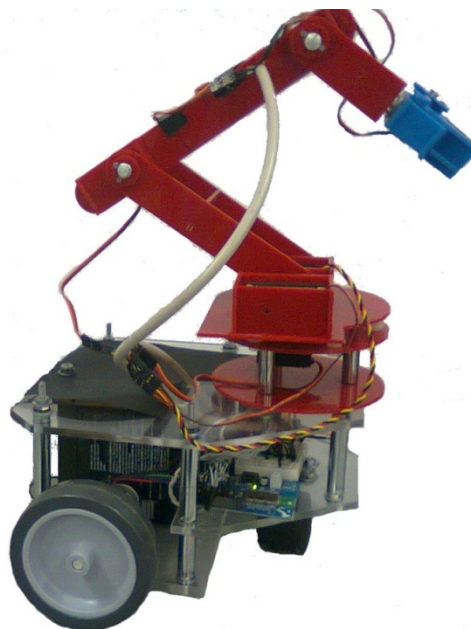


Figura III – 21. Prototipo físico del Manipulador Móvil diseñado.

Finalmente, cabe aclarar que ésta, es la primera iteración en un proceso de diseño mucho más largo. El prototipo requirió de ciertos ajustes que no se consideraron al momento de diseñarlo, detalles que no interfirieron con su funcionalidad. De cualquier modo, se corrigieron manualmente, para realizar el presente trabajo.

IV. MODELADO CINEMÁTICO.

En el capítulo presente se desarrolla la cinemática del Manipulador Móvil, ésta se tratará por separad: brazo manipulador, plataforma móvil. Además, en la última sección del capítulo, se menciona el acoplamiento encargado de coordinar ambas partes. Como se lee en el título del capítulo, sólo se aborda el tema de la cinemática, si se tiene interés sobre la dinámica de cada sistema se recomienda acudir a las referencias [18], [19] y [20].

IV. 1. MANIPULADOR SERIAL.

La cinemática de un manipulador serial expone el estudio analítico de la geometría del movimiento, respecto a un sistema de coordenadas de referencia fijo, como una función del tiempo, sin considerar las fuerzas o momentos que originan el movimiento. De manera que la cinemática, se interesa por la descripción analítica del desplazamiento espacial del robot, en particular entre la posición de las variables de articulación, la posición y orientación del efector final del brazo manipulador.

Existen dos problemas fundamentales en la cinemática del manipulador. El primer problema se le conoce como cinemática directa, mientras que el segundo es el problema es de cinemática inversa.

IV. 1. 1. CINEMÁTICA DIRECTA.

Se utiliza algebra vectorial y matricial, para desarrollar un método generalizado y sistemático, que describa y represente la localización de los elementos de un manipulador respecto a un sistema de referencia fijo. Como los eslabones de un manipulador pueden girar y/o trasladarse, con respecto a un sistema de coordenadas de referencia, entonces, se establece un sistema de coordenadas ligado al eslabón a lo largo del eje de la articulación, para cada uno. El problema cinemático directo consiste en, encontrar una matriz de transformación que relaciona el sistema de coordenadas, relacionado al eslabón, al sistema de coordenadas de referencia. La nomenclatura que se utiliza en esta sección, fue tomada de J. Ángeles [17].

IV. 1. 1. 1. NOTACION DE DENAVIT – HARTENBERG.

La forma más simple de modelar geoméricamente un robot manipulador, es aplicando el concepto de cadena cinemática. Una cadena cinemática es un conjunto de cuerpos rígidos (también llamados eslabones), acoplados por pares

cinemáticos (también denominados juntas), entonces se tiene, un par cinemático es el acoplamiento de dos cuerpos rígidos con la finalidad de restringir su movimiento relativo. Existen dos tipos básicos de pares cinemáticos: de alto orden y de bajo orden. Un par de bajo orden cinemático, ocurre cuando el lugar de contacto es a lo largo de una superficie común de dos cuerpos. Hay seis diferentes de pares, pero todos pueden ser producidos desde dos tipos básicos, esto es, el par rotacional, denotado por R (llamado de revolución), y el par deslizante, representado por P (llamado prismático).

Con el propósito de describir la arquitectura de una cadena cinemática, es decir, describir la posición y orientación de los ejes de sus juntas, es que se introduce la notación de Denavit – Hartenberg.

Para lograr la notación de Denavit – Hartenberg primero se realiza lo siguiente: los eslabones se numeran $0, 1, \dots, n$, el i – ésimo par es definido como el que acopla el $(i - 1)$ – ésimo eslabón con es i – ésimo eslabón. Por lo tanto, se asume que el manipulador está compuesto de $n + 1$ eslabones y n pares. El eslabón 0 , es la base fija, mientras que el eslabón n es el efector final. Luego un marco coordinado F_i , se define con el origen O_i y los ejes X_i, Y_i, Z_i . Este marco se adjunta al $(i - 1)$ – ésimo eslabón (no al i – ésimo eslabón), para $i = 1, \dots, n + 1$. Ésta es la clásica notación de Denavit – Hartenberg. Para los primeros n marcos, esto se hace siguiendo las siguientes reglas:

1. Z_i es el eje del i – ésimo par. Se advierte que, hay dos posibilidades de definir la dirección positiva de este eje. Ya que cada eje par es solo una línea, no un segmento dirigido. Por otra parte, el eje Z_i de un par prismático puede ser colocado arbitrariamente, pues sólo se define su dirección.
2. X_i se define como la perpendicular común a Z_{i-1} y Z_i , dirigida de la primera a la segunda. Notando que si estos dos ejes se intersectan, la dirección positiva de X_i este indefinida y, por lo tanto, puede ser libremente asignado.
3. La distancia entre Z_i and Z_{i+1} se define como a_i , que es, por lo tanto, no-negativa.
4. La coordenada en Z_i de la intersección de Z_i con X_{i+1} se denota por b_i . Debido a que esta cantidad es una coordenada, puede ser positiva o negativa. Su valor absoluto es la distancia entre X_i y X_{i+1} , también llamado, desplazamiento entre las perpendiculares sucesivas comunes, a los correspondientes ejes de las juntas.
5. El ángulo entre Z_i y Z_{i+1} se define como α_i y es medida respecto a la dirección positiva de X_{i+1} . Este parámetro es conocido como, el ángulo de giro entre los ejes pares sucesivos.
6. El ángulo entre X_i y X_{i+1} se define como θ_i y es medido respecto a la dirección positiva de Z_i .

En resumen, un manipulador de $n - \text{ ejes}$ está compuesto de $n + 1$ eslabones y $n + 1$ marcos coordenados.

Siguiendo las reglas que se acaban de presentar, para realizar la notación de Denavit – Hartenberg (DH), se muestra el resultado del proceso mencionado arriba, en la Figura IV – 1, aplicado el método al manipulador que se diseñó. De igual forma en la Tabla IV – 1 se anotan las magnitudes físicas, que estos parámetros representan, en este tipo de manipulador. Es muy importante destacar, que los valores de estos parámetros solo se aplican para este manipulador, cualquier otro manipulador incluso siendo de cinco GDL sus parámetros son diferentes.

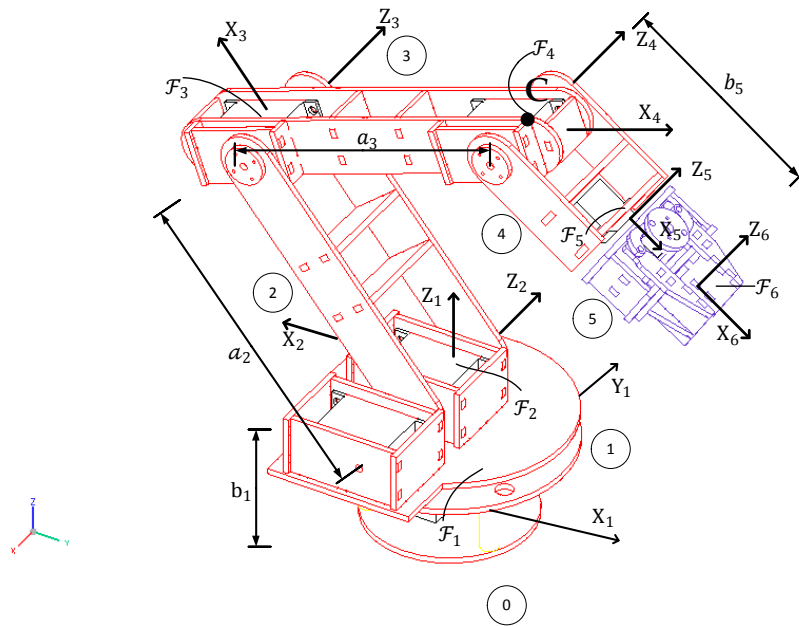


Figura IV – 1. Notación de Denavit – Hartenberg para manipulador diseñado.

i	a_i	b_i	α_i	θ_i
1	0	9.3 cm	90°	$\theta_1 + 180^\circ$
2	15 cm	0	0	θ_2
3	15 cm	0	0	θ_3
4	0	0	90°	$\theta_4 + 90^\circ$
5	0	11.7 cm	0	θ_5

Tabla IV – 1. Parámetros de Denavit –Hartenberg para manipulador diseñado.

El último marco, está definido de manera arbitraria, pero su origen es colocado en un punto en específico del EE, este punto se le conoce como punto de operación p , el cual, define la tarea de la mano. Además, los dos últimos ejes se intersectan en el punto C , por lo tanto, los puntos, de los últimos dos eslabones, se mueven en una esfera concéntrica. La subcadena que se forma se conoce como una muñeca esférica, con C como centro. De la misma manera, la subcadena compuesta de los primeros cuatro eslabones, es llamada brazo. En

consecuencia, se tiene que la muñeca está desacoplada del brazo, y es usada para propósitos de orientación. El brazo se usa para posicionar al punto C . A este manipulador se le conoce como, del tipo desacoplado.

IV. 1. 1. 2. INTERPRETACIÓN GEOMETRICA DE MANIPULADOR

La posición y orientación relativa entre los eslabones está totalmente determinada por:

- i) La matriz de rotación tomando los ejes X_i, Y_i, Z_i en la configuración en la cual, están en paralelo a los ejes $X_{i+1}, Y_{i+1}, Z_{i+1}$.
- ii) El vector de posición del origen de i a $i + 1$.

Primero, se encuentra la representación matricial de la rotación Q_i llevando \mathcal{F}_i a la orientación coincidente con la de \mathcal{F}_{i+1} , asumiendo, sin perder generalidad, que los dos orígenes son coincidentes. Esta matriz, es más fácilmente obtenida si la rotación de interés se descompone en dos rotaciones sucesivas. Estas rotaciones son:

$$[C_i]_i = \begin{bmatrix} \cos \theta_i & -\sin \theta_i & 0 \\ \sin \theta_i & \cos \theta_i & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad [\Lambda_i]'_i = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha_i & -\sin \alpha_i \\ 0 & \sin \alpha_i & \cos \alpha_i \end{bmatrix} \quad (4.1)$$

Por lo tanto, la matriz buscada se calcula por:

$$[Q_i]_i = Q_i = [C_i]_i [\Lambda_i]'_i \quad (4.2)$$

Con el fin de encontrar la expresión para el vector de posición \mathfrak{a}_i , que conecta a los orígenes de los marcos de referencia i con el $i + 1$, dirigido del primero al segundo, se tiene lo siguiente:

$$\mathfrak{a}_i = \overrightarrow{O_i O_{i+1}} = \overrightarrow{O_i O_{i'}} + \overrightarrow{O_{i'} O_{i+1}} \quad (4.3)$$

Donde cada uno representa:

$$[\overrightarrow{O_i O_{i'}}]_i = \begin{bmatrix} 0 \\ 0 \\ b_i \end{bmatrix}, \quad [\overrightarrow{O_{i'} O_{i+1}}]_{i+1} = \begin{bmatrix} a_i \\ 0 \\ 0 \end{bmatrix} \quad (4.4)$$

De forma que para encontrar la expresión de la ecuación 4.3 los dos vectores se deben de expresar en el mismo marco coordenado, teniendo:

$$[\overrightarrow{O_{i'} O_{i+1}}]_i = [Q_i]_i [\overrightarrow{O_{i'} O_{i+1}}]_{i+1} = \begin{bmatrix} a_i \cos \theta_i \\ a_i \sin \theta_i \\ 0 \end{bmatrix} \quad (4.5)$$

Y por consiguiente:

$$[\mathfrak{a}_i]_i = \mathfrak{a}_i = \begin{bmatrix} a_i \cos \theta_i \\ a_i \sin \theta_i \\ b_i \end{bmatrix} \quad (4.6)$$

Considerando lo anterior, si se quiere obtener el vector de posición p de un punto operacional P del EE, se tiene que encontrar la siguiente expresión:

$$[\mathfrak{a}_1]_1 + [\mathfrak{a}_2]_1 + [\mathfrak{a}_3]_1 + [\mathfrak{a}_4]_1 + [\mathfrak{a}_5]_1 = [p]_1 \quad (4.7)$$

La ecuación de arriba, requiere que todos los vectores involucrados, se expresen en el mismo marco coordinado. Como se mencionó anteriormente, es conveniente representar para cada marco coordinado, las relaciones sucesivas. Estas relaciones se pueden realizar mediante transformaciones de similitud, teniendo esto en cuenta, la ecuación (4.7) se vuelve:

$$\mathbf{a}_1 + Q_1(\mathbf{a}_2 + Q_2\mathbf{a}_3 + Q_2Q_3\mathbf{a}_4 + Q_2Q_3Q_4\mathbf{a}_5) = p \quad (4.8)$$

En donde ambos lados de la ecuación de arriba, se encuentran en el mismo marco coordinado y con esta ecuación se puede encontrar el punto operacional referido a la base inercial.

IV. 1. 2. CINEMÁTICA INVERSA.

Los robots controlados por computadora, suelen utilizarse en el espacio de las variables de juntas, mientras que los objetos que se manipulan, suelen expresarse en las coordenadas del mundo. Con el fin de controlar la posición y la orientación del efector final de un manipulador para realizar su tarea, es más importante encontrar la solución a la cinemática inversa. Dicho de otra forma, con la posición y orientación del efector final del manipulador y los parámetros de cada junta, se desea encontrar el valor de los ángulos para cada junta del manipulador y así realizar la tarea.

Existen varias formas de resolver el problema cinemático inverso, de estos métodos, se utiliza en este trabajo un método geométrico descrito por Fu et. al. [18], este método tiene la ventaja de encontrar los valores de las juntas en línea, permitiendo que cualquier alteración en la posición del efector final del manipulador se pueda alcanzar. Esto se debe a que no requiere gran procesamiento para calcular las variables requeridas.

Utilizando este método, se tiene principalmente dos soluciones para este problema, estas son: codo arriba y codo abajo. Para este trabajo, se utiliza la configuración codo arriba. Esta elección es de acuerdo a la tarea que va a desempeñar, permitiendo que los objetos encontrados debajo del origen del manipulador, puedan ser alcanzados. Lo anterior, se debe a que los servomotores con los que trabaja poseen un rango de movimiento limitado, es decir, no alcanzar a describir una vuelta completa, solo la mitad de ella.

Antes de desarrollar el proceso, es importante destacar un detalle. Regularmente, este proceso lleva a encontrar los ángulos de cada junta, mediante la ley de cosenos y la ley de senos, lo que lleva a que las expresiones a las que se llega, implican encontrar el arco coseno de alguna expresión. Esta solución es inconsistente y mal acondicionada, porque de la función arco coseno no se obtiene una buena precisión en la determinación del ángulo, se debe a que $\cos(\theta) = \cos(-\theta)$.

Por lo que, se tiene que encontrar una función trigonométrica más consistente para la solución del problema cinemático inverso. Para resolver esta cuestión se utiliza $ang\ tan2(y, x)$, que devuelve el ángulo ajustado a su propio cuadrante, se define de la siguiente forma:

$$\theta = ang\ tan2(y, x) = \begin{cases} 0^\circ \leq \theta \leq 90^\circ & \text{para } +x, +y \\ 90^\circ \leq \theta \leq 180^\circ & \text{para } -x, +y \\ -180^\circ \leq \theta \leq -90^\circ & \text{para } -x, -y \\ -90^\circ \leq \theta \leq 0^\circ & \text{para } +x, -y \end{cases} \quad (4.9)$$

Con la ecuación 4.9, se obtiene el valor de los ángulos de las juntas. Desacoplando al manipulador en el problema de posicionamiento y el problema de orientación, se puede abordar de forma separada. Con lo que se puede encontrar primero la expresión, para obtener los ángulos de las primeras tres juntas y finalmente las últimas dos. Para el problema de posicionamiento del manipulador, se debe obtener los valores para el punto C , que se muestra en la Figura IV – 1, el valor en el tiempo de este punto se muestra en la sección donde se resuelve el problema de orientación. Lo mencionado es correcto en todo momento, pues el punto C se encuentra, en todo momento, en el mismo plano que el punto p del EE. El manipulador diseñado no cuenta con la tercera junta de orientación, que cambia el plano en el que se encuentra este punto.

IV. 1. 2. 1. PROBLEMA DE POSICIÓN.

Solución de la primera junta: gracias a que el punto p y C se encuentran en el mismo plano $x - z$, el ángulo de la primera junta se puede encontrar con las coordenadas (x, y) del punto p , siendo estas coordenadas referidas al marco de origen del manipulador, como se ve en la Figura IV – 2. De esta forma la primera junta se encuentra con la expresión:

$$\theta_1 = ang\ tan\left(\frac{y_c}{x_c}\right) \quad (4.10)$$

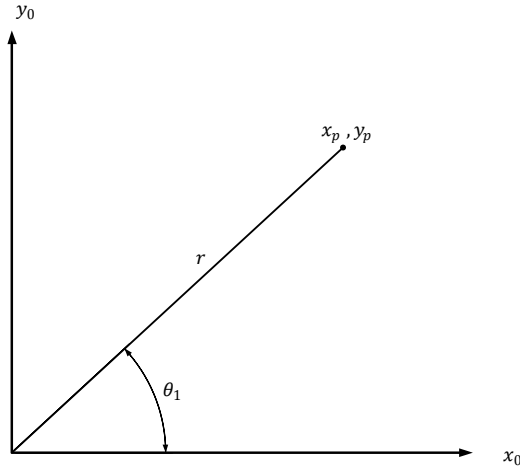


Figura IV – 2. Solución para la primera junta.

Solución a la segunda junta: para resolver las siguientes dos juntas, se proyecta el manipulador en el plano $x_0 - z_0$, como se muestra en la Figura IV – 3. Con la proyección del manipulador en este plano, se puede ver que la solución de la segunda junta está dada por:

$$\theta_2 = \pi - \alpha - \beta \quad (4.11)$$

Como se muestra en la ecuación 4.9, se tienen que encontrar las expresiones para los ángulos, en función del ángulo cuyo tangente, con esto se tiene que:

$$\tan \theta_2 = \frac{\sin \theta_2}{\cos \theta_2} \quad (4.12)$$

Por consiguiente:

$$\theta_2 = \text{ang} \tan \frac{\sin \theta_2}{\cos \theta_2} \quad (4.13)$$

Con la ecuación 4.13, se aprecia que hay que encontrar las soluciones respecto al seno del ángulo de la junta y al coseno del ángulo de la junta. Por lo que, para encontrar estas relaciones se utilizan identidades trigonométricas y ley de cosenos. Teniendo en consideración lo anterior, se tiene entonces que:

$$\sin \theta_2 = \sin(\pi - \alpha - \beta) \quad (4.14)$$

Utilizando la identidad trigonométrica del seno de la suma y resta de dos ángulos, se obtiene lo siguiente:

$$\sin \theta_2 = \sin(\alpha + \beta) \quad (4.15)$$

Aplicando nuevamente la identidad de suma de dos ángulos se tiene:

$$\sin \theta_2 = (\sin \alpha \cos \beta + \cos \alpha \sin \beta) \quad (4.16)$$

Para obtener el coseno del ángulo de la junta dos, se realiza lo mismo que para el caso del seno, la única diferencia es que se utilizan las correspondientes identidades para el coseno, obteniendo la siguiente expresión:

$$\cos \theta_2 = -(\cos \alpha \cos \beta - \sin \alpha \sin \beta) \quad (4.17)$$

Las expresiones 4.16 y 4.17, muestran productos de senos y cosenos de los ángulos que conforman a θ_2 , por lo que para encontrar este ángulo, se obtienen estas expresiones para α y β . De la Figura IV – 3 se obtiene:

$$\sin \alpha = \frac{z_c - l_1}{R} \quad , \quad \cos \alpha = \frac{x_c}{R} \quad (4.18)$$

Para encontrar las expresiones de β , se utiliza la ley de cosenos, y para obtener el seno, se utiliza la identidad pitagórica, consiguiendo lo siguiente:

$$l_3^2 = R^2 + l_2^2 - 2 l_2 R \cos \beta$$

$$\cos \beta = \frac{R^2 + l_2^2 - l_3^2}{2 l_2 R} \quad , \quad \sin \beta = \sqrt{1 - \cos^2 \beta} \quad (4.19)$$

Donde para 4.18 y 4.19, R se obtiene de la Figura IV – 3 y está dada por la expresión:

$$R = \sqrt{x_c^2 + (z_c - l_1)^2} \quad (4.20)$$

Por lo tanto, de acuerdo a la expresión 4.13, teniendo en cuenta 4.16 y 4.17 y además considerando la expresión 4.9 la solución de la junta dos está dada por:

$$\theta_2 = \text{ang tan2} (\sin \alpha \cos \beta + \cos \alpha \sin \beta, -(\cos \alpha \cos \beta - \sin \alpha \sin \beta)) \quad (4.21)$$

Finalmente sustituyendo en 4.21 las expresiones 4.18 y 4.19 se tiene:

$$\theta_2 = \text{ang tan2} \left(\frac{z_c - l_1}{R} \frac{R^2 + l_2^2 - l_3^2}{2 l_2 R} + \frac{x_c}{R} \sqrt{1 - \cos^2 \beta}, -\left(\frac{x_c}{R} \frac{R^2 + l_2^2 - l_3^2}{2 l_2 R} - \frac{z_c - l_1}{R} \sqrt{1 - \cos^2 \beta} \right) \right) \quad (4.22)$$

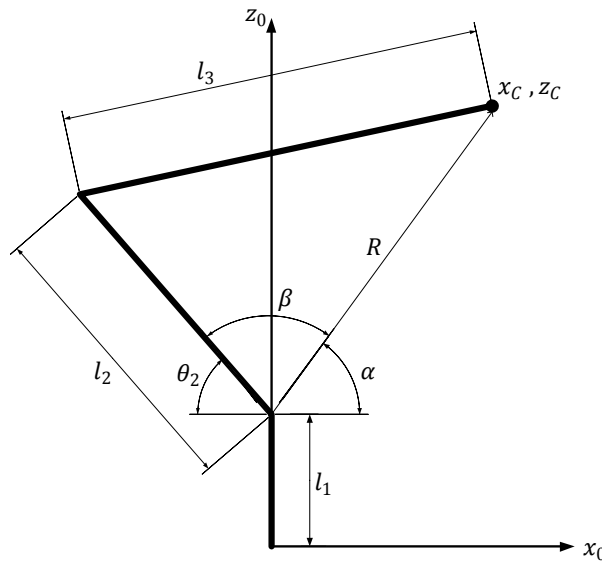


Figura IV – 3. Solución para la segunda junta.

Solución para la tercera junta: de forma similar a la junta anterior, se proyecta el manipulador en el plano $x_0 - z_0$ como se muestra en la Figura IV – 4. En esta figura se encuentra, que el ángulo de la junta tres está dado por la relación:

$$\theta_3 = \pi - \gamma \quad (4.23)$$

De igual forma que con la junta dos, se tiene que encontrar el ángulo de la junta tres, mediante la expresión ángulo cuya tangente, como se muestra en la expresión 4.24. Por lo tanto, se tiene que encontrar las expresiones de seno y coseno del ángulo que esta expresión determina.

$$\theta_3 = \text{ang tan} \frac{\sin \theta_3}{\cos \theta_3} \quad (4.24)$$

Aplicando el seno y el coseno a la ecuación 4.23 y nuevamente aplicando las identidades de suma y resta de ángulos de senos y cosenos, se obtienen las siguientes expresiones:

$$\sin \theta_3 = \sin(\pi - \gamma) = \sin \gamma \quad (4.25)$$

$$\cos \theta_3 = \cos(\pi - \gamma) = -\cos \gamma \quad (4.26)$$

Lo que muestran las expresiones 4.25 y 4.26 es que se deben de encontrar las relaciones de seno y coseno para el ángulo gamma. Por consiguiente, se utiliza la ley de cosenos para encontrar el coseno del ángulo y la identidad pitagórica para encontrar el seno del mismo. El resultado se muestra en las siguientes expresiones como se ve en la Figura IV – 4:

$$R^2 = l_3^2 + l_2^2 - 2 l_2 l_3 \cos \gamma$$

$$\cos \gamma = \frac{l_3^2 + l_2^2 - R^2}{2 l_2 l_3}, \quad \sin \gamma = \sqrt{1 - \cos^2 \gamma} \quad (4.27)$$

Los parámetros de las anteriores expresiones, se muestran en la Figura IV – 4 y la relación de R se describe en 4.20. Por lo tanto, teniendo en consideración la expresión 4.24, las expresiones 4.25 y 4.26, y además, expresando el resultado como se muestra en la expresión 4.9, se tiene que la solución de la junta tres está dada por:

$$\theta_3 = \text{ang tan2} (\sin \gamma, -\cos \gamma) \quad (4.28)$$

Finalmente, sustituyendo las expresiones 4.27 en la ecuación 4.28, se tiene la solución para la tercera junta:

$$\theta_3 = \text{ang tan2} \left(\sqrt{1 - \cos^2 \gamma}, -\frac{l_3^2 + l_2^2 - R^2}{2 l_2 l_3} \right) \quad (4.29)$$

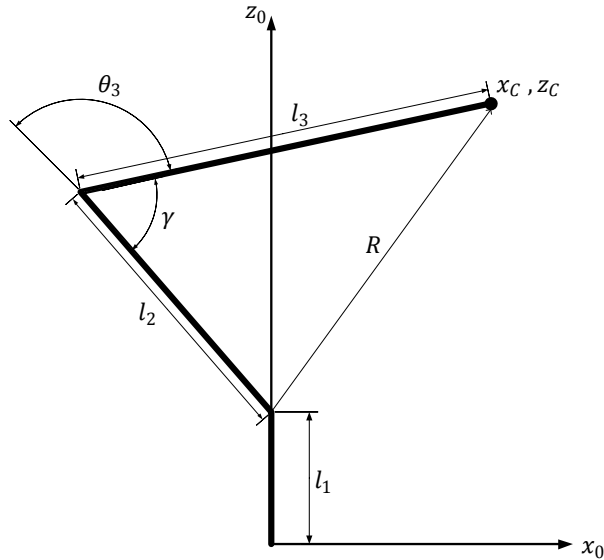


Figura IV – 4. Solución para la tercera junta.

IV. 1. 2. 2. PROBLEMA DE ORIENTACIÓN.

El problema de orientación va dirigido a resolver el ángulo de las últimas dos juntas, para realizar la tarea, esto quiere decir, que la solución dependerá del objeto a manipular. Antes de continuar con el problema de orientación, se tiene que determinar el punto C . Como ya se estipuló anteriormente, el usuario o el programa, determina el punto en el espacio global, al que el punto p del efector final debe de llegar. Quiere decir que, el punto C no es la entrada para el sistema, pero si es el medio para encontrar la solución al problema de posición del manipulador. Por consiguiente, se tiene que encontrar el punto C a través de la entrada, que es el punto p o punto de operación. Con lo anterior mencionado, en la Figura IV – 5 se muestra un diagrama para determinar el valor del punto C en su coordenada z .

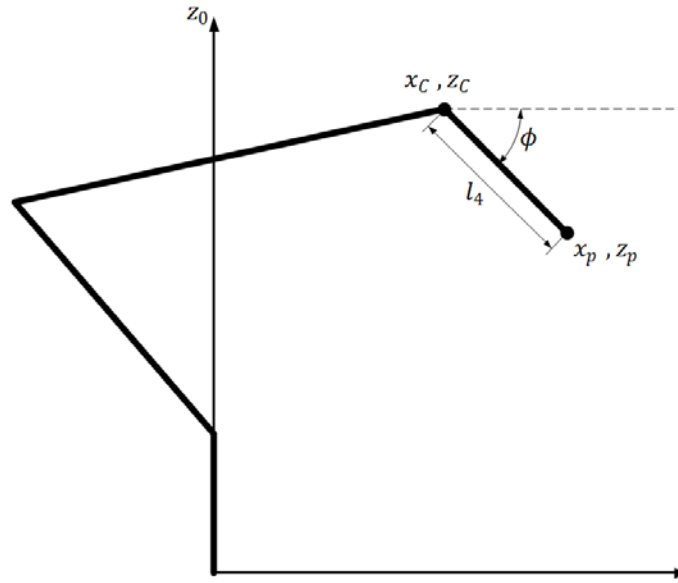


Figura IV – 5. Cálculo de la coordenada z del punto C del manipulador.

De la figura de arriba, se encuentra la coordenada z y x del punto C mediante las siguientes expresiones:

$$z_c = z_p + l_4 \sin \phi \quad (4.30)$$

$$x_c = x_p - l_4 \cos \phi \quad (4.31)$$

La coordenada y, no es necesario obtenerla, debido a que sólo se necesita para obtener el ángulo de la primera junta, esto por encontrarse en el mismo plano. En el caso de encontrarse la última junta de orientación, ya no se requiere calcular esta coordenada.

La ecuación 4.31, funciona correctamente cuando se encuentra el manipulador en este plano, es decir, el ángulo de la junta uno es cero, como no siempre es cero, la forma de corregir esta ecuación se muestra en seguida. Se soluciona mediante la magnitud de un vector, que va del origen al punto p. Este vector se ve en la Figura IV – 2, que corresponde a la dimensión, en el plano en el cual el desplazamiento angular de la junta uno se desprecia, teniendo lo siguiente:

$$r = \sqrt{x_p^2 + y_p^2}$$

$$x_c = r - l_4 \cos \phi \quad (4.32)$$

En la figura de arriba y las dos ecuaciones de arriba se encuentra un parámetro ϕ , este parámetro corresponde a la orientación del objeto a manipular, y es el ángulo respecto a la horizontal con el eslabón cuatro, de longitud l_4 . Este ángulo dependerá de cómo se quiera manipular el objeto, es decir, en qué ángulo respecto a la horizontal será tomado.

Solución para la cuarta junta: al igual que las juntas anteriores se proyecta el manipulador en el plano $x_0 - z_0$ como se muestra en la Figura IV – 6. En esta

figura se aprecia que la relación para determinar el ángulo de la cuarta junta, está dada por:

$$\theta_4 = \phi + \gamma - \theta_2 \quad (4.33)$$

El ángulo de la junta cuatro, tiene que ser encontrado mediante la función trigonométrica de ángulo cuya tangente. Como muestra la expresión 4.9, esto quiere decir, que hay que aplicar el seno y el coseno a la expresión 4.33, obteniendo lo siguiente:

$$\sin \theta_4 = \sin(\phi + \gamma - \theta_2) \quad , \quad \cos \theta_4 = \cos(\phi + \gamma - \theta_2) \quad (4.34)$$

Aplicando a las expresiones 4.34, sus correspondientes identidades trigonométricas de suma y resta de ángulos, se tiene, las siguientes expresiones:

$$\sin \theta_4 = \sin \phi (\cos \gamma \cos \theta_2 + \sin \gamma \sin \theta_2) + \cos \phi (\sin \gamma \cos \theta_2 - \cos \gamma \sin \theta_2) \quad (4.35)$$

$$\cos \theta_4 = \cos \phi (\cos \gamma \cos \theta_2 + \sin \gamma \sin \theta_2) - \sin \phi (\sin \gamma \cos \theta_2 - \cos \gamma \sin \theta_2) \quad (4.36)$$

Siendo estas dos últimas ecuaciones, las necesarias para encontrar la solución de la junta cuatro, los factores seno y coseno para γ y θ_2 se encuentran en la expresión 4.27 para γ . Como el ángulo para la junta dos (θ_2) ya se encontró previamente sólo se aplican las correspondientes identidades trigonométricas para el valor numérico. Tomando las expresiones 4.9 así como las 4.35 y 4.36 se encuentra la solución para la cuarta junta:

$$\text{ang tan2} (\sin \phi (\cos \gamma \cos \theta_2 + \sin \gamma \sin \theta_2) + \cos \phi (\sin \gamma \cos \theta_2 - \cos \gamma \sin \theta_2), \cos \phi (\cos \gamma \cos \theta_2 + \sin \gamma \sin \theta_2) - \sin \phi (\sin \gamma \cos \theta_2 - \cos \gamma \sin \theta_2)) \quad (4.37)$$

Considerando, que se tienen que sustituir los términos en función de ϕ y γ , por las expresiones en 4.27 y 4.34. Por cuestiones de espacio no se sustituyen aquí, pero la idea queda aclarada.

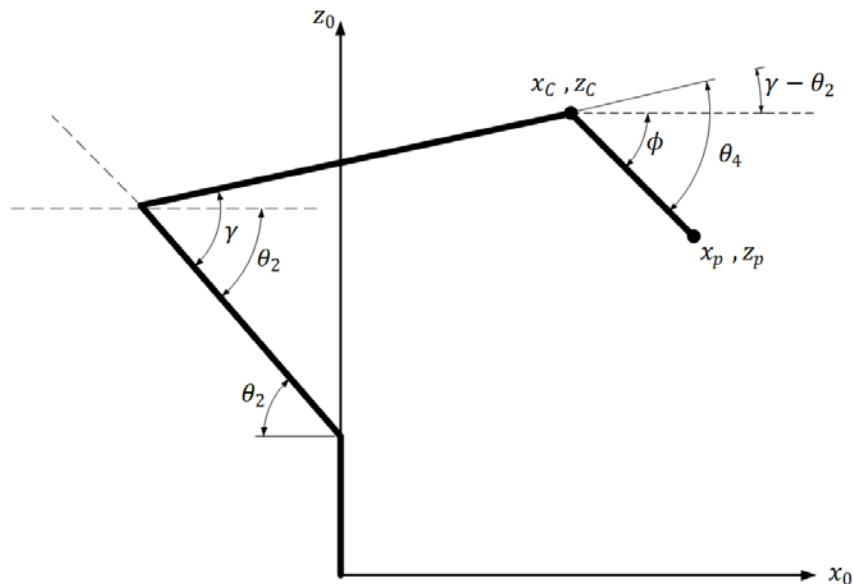


Figura IV – 6. Solución para la cuarta junta.

Solución para la quinta junta: así como la junta anterior es para orientar el efector final respecto al objeto a manipular, se tiene el mismo caso para esta junta.

Para encontrar el ángulo de la quinta junta, es necesario primero aclarar algunas cuestiones sobre la ubicación del objeto a manipular, debido a que estas cuestiones tienen que ser primeramente descritas, no se podrá incluir en este capítulo. Por lo que, para el momento solo se tendrá que:

$$\theta_5 = \psi \quad (4.38)$$

Lo anterior, para no confundir al lector con detalles que no se han explicado aún y que no es oportuno mencionar en el presente capítulo. Se adelanta, que estos detalles tienen que ver con la ubicación del manipulador móvil en el espacio, la ubicación del objeto a manipular y otras cuestiones sobre orientación. Todo lo anterior se trata en el siguiente capítulo.

Con todo lo anterior mencionado, en esta sección se logran obtener las expresiones para cada junta, estos son de suma importancia, ya que solo serán necesarias introducir las coordenadas del punto p , para calcular el ángulo de todas las juntas del manipulador, a través de las ecuaciones 4.10, 4.22, 4.29, 4.37 y 4.38.

IV. 2. PLATAFORMA MÓVIL.

Una plataforma móvil con ruedas, se asume que consiste de un cuerpo rígido, no deformable y en la que las llantas no presentan deslizamientos, además, de que se mueve en un plano horizontal. Las ruedas, en todo momento, se mantienen verticales y giran a lo largo del eje del motor, acoplado a ellas. Todas las notaciones y nomenclatura que se utiliza en esta sección, se encuentran en el documento de V. J. González [20], [21].

La plataforma móvil con ruedas, cuenta con dos marcos coordenados para su análisis, como se muestra en la Figura IV – 7. El marco inercial o marco global se representa por $x - y$ fijados al punto origen (0). En este marco, se encuentra todo el entorno en el cual se moverá la plataforma. El otro marco, es el de la plataforma que se encuentra en movimiento junto con el robot. Esto representado por $x_1 - y_1$ fijado al punto P que se localiza en el centroide de la placa de transporte de la plataforma móvil. Entonces, con este marco se puede describir la postura del robot móvil por el siguiente vector:

$$\xi = [x \ y \ \theta]^T \quad (4.39)$$

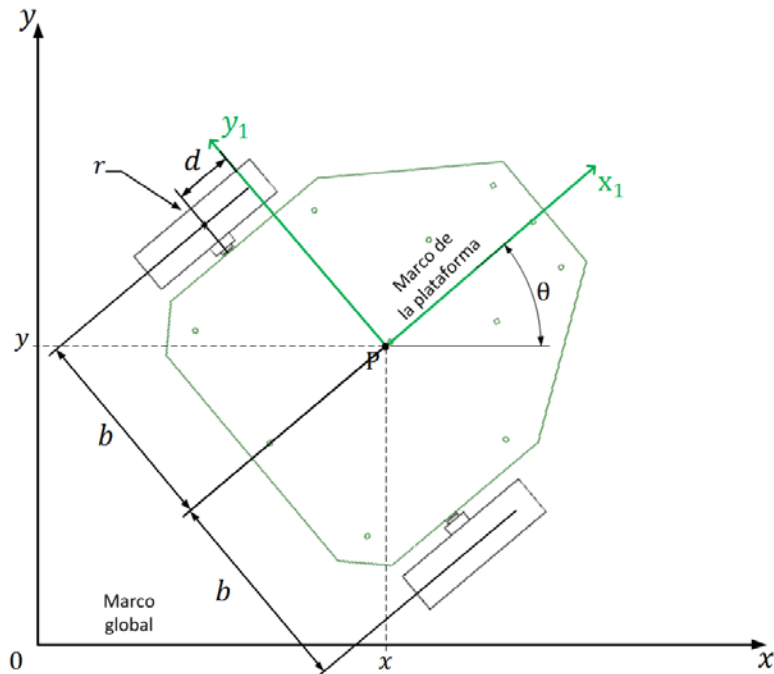


Figura IV – 7. Postura de la Plataforma móvil.

Este vector describe la posición del punto $P = (x, y)$ y la orientación θ del marco de la plataforma $x_1 - y_1$, referente al marco global $x - y$. Una situación similar a la del manipulador, se encuentra en la plataforma móvil, pues el ángulo θ se mide del eje x al x_1 . Es necesario obtener una matriz de rotación de la postura, que consiste de una matriz de rotación ortonormal que mapea del marco global al marco de la plataforma móvil, y se describe mediante la siguiente expresión:

$$R(\theta) = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.40)$$

IV. 2. 1. PROPAGACIÓN DE VELOCIDADES

Para encontrar las restricciones cinemáticas de la plataforma móvil, se utiliza la técnica de propagación de velocidades de los eslabones, que es descrito por J. Craig [19]. Para aplicar esta técnica, se sugiere observar el diagrama de la Figura IV – 8, además, estará descrita en la presente sección.

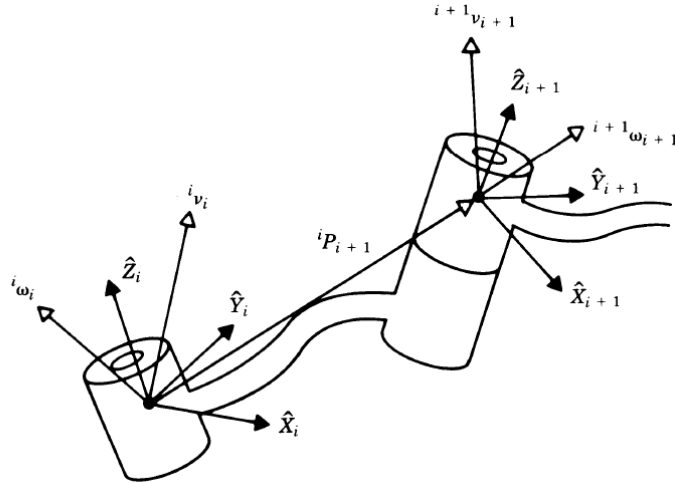


Figura IV – 8. Vectores de velocidad de los eslabones colindantes [19].

La velocidad del eslabón $i + 1$, es la del eslabón i más cualquier otra componente de velocidad que agregue la junta $i + 1$. Las velocidades de rotación se agregan cuando los vectores ω se trazan en el mismo marco. Por lo que la velocidad angular del eslabón $i + 1$, es la misma que la del eslabón i , y en la que agrega un nuevo término producido por la velocidad de rotación de la junta $i + 1$. Y así queda expresado en el marco de referencia i :

$${}^i\omega_{i+1} = {}^i\omega_i + {}_{i+1}^iR\dot{\theta}_{i+1} {}^{i+1}Z_{i+1} \quad (4.41)$$

Donde:

$$\dot{\theta}_{i+1} {}^{i+1}Z_{i+1} = \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_{i+1} \end{bmatrix} \quad (4.42)$$

Si se pre-multiplican ambos lados de la ecuación 4.41 por ${}^{i+1}_iR$, se encuentra la expresión que describe la velocidad angular del eslabón $i + 1$ en el marco $i + 1$, teniendo:

$${}^{i+1}\omega_{i+1} = {}^{i+1}_iR {}^i\omega_i + \dot{\theta}_{i+1} {}^{i+1}Z_{i+1} \quad (4.43)$$

Por otro lado, la velocidad lineal del origen del marco $i + 1$, es la misma que la del origen del marco i agregándole un nuevo término, producido por la velocidad de rotación del eslabón i , teniendo:

$${}^i v_{i+1} = {}^i v_i + {}^i\omega_i \times {}^i P_{i+1} \quad (4.44)$$

Donde ${}^i P_{i+1}$ es vector constante en el marco i , que indica la coordenada del origen del marco $i + 1$ respecto al marco i . Pre-multiplicando ambos lados de la ecuación 4.44 por ${}^{i+1}_iR$, se tiene:

$${}^{i+1}v_{i+1} = {}^{i+1}_iR({}^i v_i + {}^i\omega_i \times {}^i P_{i+1}) \quad (4.45)$$

Las ecuaciones 4.43 y 4.45 son válidas, cuando la junta es rotacional. Si se aplican las anteriores ecuaciones de eslabón a eslabón se pueden obtener las velocidades angulares y lineales del eslabón, en su mismo marco de referencia.

Esto, es de suma importancia para obtener las restricciones cinemáticas de la plataforma móvil, como se ve en la siguiente sección.

IV. 2. 2. RESTRICCIONES CINEMÁTICAS.

Tomando en cuenta todo lo descrito en la sección anterior, se pueden obtener las restricciones cinemáticas propagando las velocidades $[\dot{x}_1 \ \dot{y}_1 \ \dot{\theta}]$ del marco de la plataforma móvil $x_1 - y_1$ al marco de las llantas. Primero, se tienen que considerar algunos aspectos, se tiene el vector de velocidad siguiente:

$$\xi_1 = [\dot{x}_1 \ \dot{y}_1 \ \dot{\theta}]^T \quad (4.46)$$

Este vector representa la postura en velocidad del marco de la plataforma móvil, respecto a ella misma, donde las velocidades lineales del punto P están representadas por \dot{x}_1 y \dot{y}_1 , y la velocidad angular por $\dot{\theta}$. Sin embargo, esta velocidad se expresa en el propio marco de la plataforma, por lo que se tiene que representar éste, en el marco global, teniendo que:

$$\dot{\xi} = R^T(\theta)\dot{\xi}_1 \quad (4.47)$$

Lo anterior, es necesario para realizar la técnica de propagación de velocidades, entonces, hay que trasladar la velocidad del punto P a las llantas. De acuerdo a la técnica y a lo que expresa la ecuación 4.47, de la Figura IV – 7, se encuentra lo siguiente:

$${}^1\omega_1 = \begin{bmatrix} 0 \\ 0 \\ \dot{\theta} \end{bmatrix} \quad (4.48)$$

$${}^1v_1 = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ 0 \end{bmatrix} = \begin{bmatrix} \dot{x} \cos \theta + \dot{y} \sin \theta \\ \dot{y} \cos \theta - \dot{x} \sin \theta \\ 0 \end{bmatrix} \quad (4.49)$$

En las ecuaciones 4.48 y 4.49, se muestran las velocidades angulares y lineales respectivamente, del marco de la plataforma, respecto a si misma. Las anteriores se necesitan para aplicar la técnica.

$${}^1P_2 = \begin{bmatrix} -d \\ -b \\ 0 \end{bmatrix} \quad (4.50)$$

$${}^1P_2 = \begin{bmatrix} -d \\ b \\ 0 \end{bmatrix} \quad (4.51)$$

Como la aplicación de esta técnica, se requiere propagar las velocidades a cada eslabón. Para el caso de la plataforma, se requiere propagar la velocidad del punto P de la plataforma a cada llanta. Por lo que la ecuación 4.50, expresa el vector de posición del punto P al centro de la llanta derecha y; la ecuación 4.51 representa el mismo vector para la llanta izquierda. Por lo tanto, hay que aplicar la técnica para ambas llantas. A continuación se representa la de la llanta derecha.

$$\begin{aligned}
{}^2v_2 &= \begin{bmatrix} \dot{x} \cos \theta + \dot{y} \sin \theta \\ \dot{y} \cos \theta - \dot{x} \sin \theta \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \dot{\theta} \end{bmatrix} \times \begin{bmatrix} -d \\ -b \\ 0 \end{bmatrix} \\
{}^2v_2 &= \begin{bmatrix} \dot{x} \cos \theta + \dot{y} \sin \theta + b\dot{\theta} \\ \dot{y} \cos \theta - \dot{x} \sin \theta - d\dot{\theta} \\ 0 \end{bmatrix} \tag{4.52}
\end{aligned}$$

En la ecuación 4.52, se describe la velocidad de la llanta derecha. Considerando que la velocidad lineal de una rueda se puede encontrar en función a su velocidad angular y el radio de la misma, la ecuación anterior se vuelve:

$$\begin{bmatrix} r \phi_d \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \dot{x} \cos \theta + \dot{y} \sin \theta + b\dot{\theta} \\ \dot{y} \cos \theta - \dot{x} \sin \theta - d\dot{\theta} \\ 0 \end{bmatrix} \tag{4.53}$$

En donde ϕ_d y r son la velocidad angular de la llanta y el radio de la misma, respectivamente. De la misma forma, que se acaba de propagar la velocidad para la llanta derecha, se propaga para la izquierda, obteniendo:

$$\begin{aligned}
{}^2v_2 &= \begin{bmatrix} \dot{x} \cos \theta + \dot{y} \sin \theta \\ \dot{y} \cos \theta - \dot{x} \sin \theta \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \dot{\theta} \end{bmatrix} \times \begin{bmatrix} -d \\ b \\ 0 \end{bmatrix} \\
{}^2v_2 &= \begin{bmatrix} \dot{x} \cos \theta + \dot{y} \sin \theta - b\dot{\theta} \\ \dot{y} \cos \theta - \dot{x} \sin \theta - d\dot{\theta} \\ 0 \end{bmatrix} \tag{4.54}
\end{aligned}$$

La velocidad de la llanta izquierda propagada, se representa por la expresión 4.54. Del mismo modo, que en la ecuación 4.53, se puede expresar la velocidad de la llanta, teniendo:

$$\begin{bmatrix} r \phi_i \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \dot{x} \cos \theta + \dot{y} \sin \theta - b\dot{\theta} \\ \dot{y} \cos \theta - \dot{x} \sin \theta - d\dot{\theta} \\ 0 \end{bmatrix} \tag{4.55}$$

De las ecuaciones 4.53 y 4.55 se obtienen las restricciones cinemáticas de la plataforma móvil. En las que la restricción cinemática, debido a que el robot se mueve en dirección del eje de simetría, y que no existe desplazamiento lateral de llantas, se representa en la ecuación 4.56. Se puede notar que corresponde a la componente en y de las ecuaciones 4.53 y 4.55.

$$\dot{y} \cos \theta - \dot{x} \sin \theta - d\dot{\theta} = 0 \tag{4.56}$$

Finalmente, se muestra en la ecuación 4.57, las restricciones cinemáticas que resultan del giro de las llantas direccionales, que corresponden a las componentes en x de las ecuaciones 4.53 y 4.55.

$$\dot{x} \cos \theta + \dot{y} \sin \theta + b\dot{\theta} = r \phi_d \quad , \quad \dot{x} \cos \theta + \dot{y} \sin \theta - b\dot{\theta} = r \phi_i \tag{4.57}$$

IV. 2. 3. COORDENADAS GENERALIZADAS DE CONFIGURACIÓN.

Existen principalmente tres tipos de llantas: las fijas, centradas direccionales y descentradas direccionales [20], una plataforma móvil que tiene un número

$N_T = N_f + N_c + N_{oc}$ de llantas convencionales, tiene un total de $n_T = 3 + N_T + N_c + N_{oc}$ coordenadas de configuración: $q = [q_1, \dots, q_{n_T}]^T$. La plataforma móvil, que se utiliza en el presente trabajo, sólo utiliza llantas fijas, es decir $N_c = 0$ y $N_{oc} = 0$. Por lo tanto, no se mencionarán las coordenadas generalizadas que otro tipo de llantas generan.

Las coordenadas generalizadas para esta plataforma móvil están formadas por:

- Las coordenadas de la postura, que corresponde a x y y que definen la posición del punto P y la orientación θ de la plataforma.
- Las coordenadas de rotación de las llantas a lo largo de su eje horizontal de cada llanta.

De esta forma, el vector de coordenadas generalizadas para la plataforma móvil en cuestión, está determinado por:

$$q = [x \quad y \quad \theta \quad \phi_d \quad \phi_i]^T \quad (4.58)$$

Por consiguiente, el vector de la postura del robot en velocidad es el siguiente:

$$\dot{q} = [\dot{x} \quad \dot{y} \quad \dot{\theta} \quad \dot{\phi}_d \quad \dot{\phi}_i]^T \quad (4.59)$$

Una vez determinadas cuales son las coordenadas generalizadas, hay que encontrar una expresión en espacio de estados para la cinemática. Seleccionando como variables de estado dos velocidades: la angular y la lineal de la plataforma móvil, que corresponden a $v(t) = [\dot{x}_1 \quad \dot{\theta}]^T = [v \quad \omega]^T$. Esto se puede realizar despejando cada coordenada generalizada y expresándola en términos de las variables de estado, el resultado de expresar en variables de estado se muestra en la ecuación 4.60.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\phi}_d \\ \dot{\phi}_i \end{bmatrix} = \begin{bmatrix} \cos \theta & -d \sin \theta \\ \sin \theta & d \cos \theta \\ 0 & 1 \\ 1/r & b/r \\ 1/r & -b/r \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (4.60)$$

IV. 3. ACOPLAMIENTO.

La forma en que se acoplan ambas partes descritas (manipulador y plataforma móvil) es la que utiliza Yamamoto en [6], la forma de acoplar se inspira en la forma de conducir un vehículo. Cuando se conduce un automóvil, el conductor “mira” a un punto o un área delante del automóvil.

Tomando en cuenta el marco coordenado que se encuentra en el centroide de la plataforma, donde x_1 está dirigido hacia delante de la plataforma, se escoge un punto P_r referenciado al marco coordenado $x_1 - y_1$ como punto de referencia. Por lo que para plataforma móvil, sus movimientos se tienen que realizar de

manera que este punto de referencia siga una trayectoria deseada. El punto de referencia se denota como $({}^1x_r, {}^1y_r)$, que está referenciado al marco $x_1 - y_1$. Por lo tanto, las coordenadas globales (x_r, y_r) del punto de referencia se encuentra mediante:

$$x_r = x + {}^1x_r \cos \theta - {}^1y_r \sin \theta \quad (4.61)$$

$$y_r = y + {}^1x_r \sin \theta + {}^1y_r \cos \theta \quad (4.62)$$

Donde x_1 y y_1 , son las coordenadas globales del punto P mencionado en la sección anterior. La selección del punto de referencia, es el acoplamiento de ambas partes. Esto lleva a que el punto de referencia mencionado, es el punto de operación p del efector final del brazo manipulador. De manera general se expresa este acoplamiento en la Figura IV – 9. El cálculo del punto de referencia se detalla en el siguiente capítulo, que habla sobre la estrategia de coordinación.

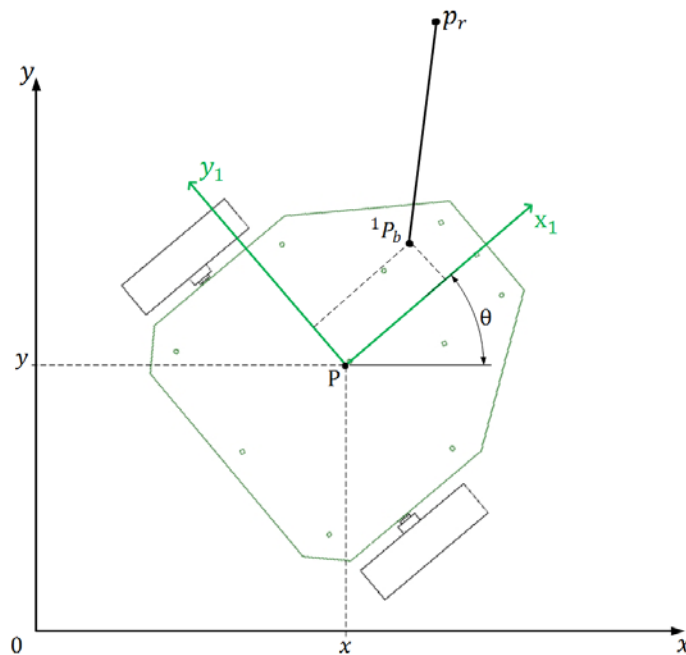


Figura IV – 9. Punto de referencia que determina el acoplamiento entre ambas partes.

V. COORDINACIÓN DE MOVIMIENTOS.

Cuando un manipulador móvil realiza una tarea, es deseable que el manipulador se encuentre en una configuración preferida, que se encuentre alejada de singularidades para que su movimiento no se vea limitado y así poder planificar el movimiento de la plataforma sin consecuencias. Por lo anterior se tiene que la estrategia utilizada es la que presenta Yamamoto en [6], donde el enfoque utilizado es por la configuración preferida. Siendo ésta la base para varios enfoques diferentes. Sin embargo no sólo se considera el enfoque de Yamamoto sino se realiza un algoritmo de decisión sin utilizar redes neuronales o inteligencia artificial, debido a su complejidad. Pero donde se implementan niveles de decisión, procesamiento y ejecución, para saber qué parte se mueve y en qué momento, similar al utilizado por Yingshu en [1]. En el presente capítulo se describe las distintas formas que se utilizaron para realizar los movimientos de ambas partes: brazo manipulador y plataforma móvil, todas las consideraciones que se realizan para lograrlo y el algoritmo propuesto para cumplir con la tarea. En la segunda parte de este capítulo se describen los sistemas computacionales utilizados así como la comunicación entre sistemas.

V. 1. PLANIFICACIÓN DE LA TRAYECTORIA.

Una técnica que es usada para llevar a la plataforma móvil a un punto requerido, al mismo tiempo que este evade obstáculos, es la técnica de campos potenciales. Este método realiza sumas vectoriales de fuerzas repulsivas virtuales, provocadas por los obstáculos y fuerzas atractivas virtuales, éstas, provocadas por los puntos de meta. La utilización de fuerzas, quiere decir que se requiere del modelo dinámico del robot, por lo que el modelo cinemático del mismo no es útil. Una forma de resolver este problema lo presenta V. J. G. Villela en [21], dónde los campos potenciales son tomados como vectores de distancia y estos se encargan de producir rangos de velocidades, lo que permite que el robot se mueva en su entorno utilizando el modelo cinemático únicamente.

V. 1. 1. CAMPOS POTENCIALES

El enfoque de campos potenciales es útil para que la plataforma móvil, se mueva de una posición a otra. Por medio de la técnica de planeación de trayectoria seleccionada, uno puede llevar al robot móvil de una postura inicial a una final. Lo anterior mediante una serie de puntos sub meta, que permita llegar a una meta final como es deseado, principalmente se debe a, la restringida movilidad de un

robot móvil, esto por el tipo de llantas utilizadas. Dentro de la técnica se ubica el problema de llegar a la meta, como se muestra en la Figura V – 1, ésta se describe a continuación.

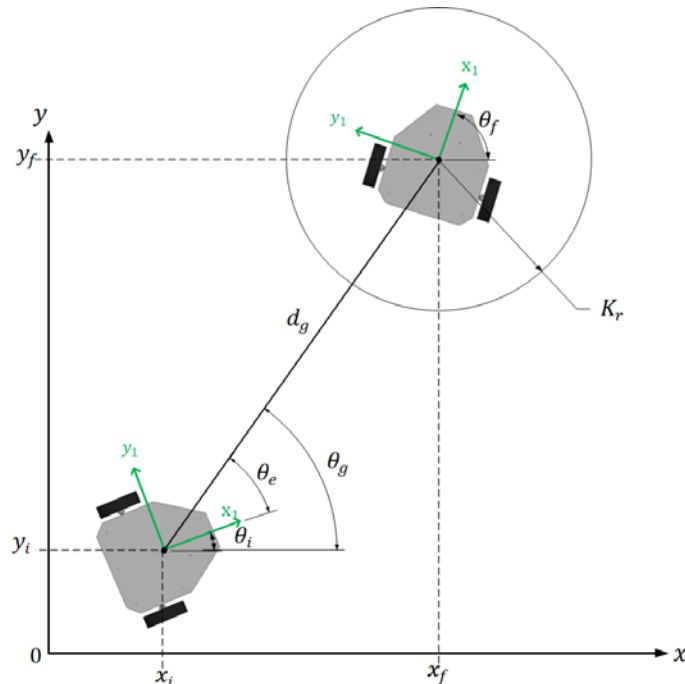


Figura V – 1. Llegando a la meta.

El problema de llegar a la meta consiste en, encontrar la velocidad y dirección a la cual el robot debe dirigirse para llegar a la meta. Para esto, primero se define un vector de distancia de la plataforma a la meta $d_g = (x_f - x_i, y_f - y_i)$, donde la magnitud de esta distancia se encuentra mediante:

$$d_g = \sqrt{(x_f - x_i)^2 + (y_f - y_i)^2} \quad (5.1)$$

Siendo el ángulo a la meta encontrado mediante:

$$\theta_g = \text{ang} \tan \frac{y_f - y_i}{x_f - x_i} \quad (5.2)$$

Y el ángulo de error se define como $\theta_e = \theta_g - \theta_i$. Entonces la plataforma móvil es llevada a la meta, tomando a las siguientes reglas a seguir.

$$v = \begin{cases} v_{max} & \text{si } |d_g| > K_r \\ \frac{v_{max}}{K_r} d_g & \text{si } |d_g| \leq K_r \end{cases} \quad (5.3)$$

Donde v_{max} es la máxima velocidad de la plataforma con la que se desplaza y K_r es el radio del área de docking. Para encontrar una expresión que determine la velocidad angular de la plataforma móvil, se tiene lo siguiente:

$$\omega = \omega_{max} \sin \theta_e \quad (5.4)$$

Donde ω_{max} es la máxima velocidad angular que la plataforma móvil puede llegar a alcanzar.

V. 1. 2. DOCKING Y PRE-DOCKING DE LA PLANEACIÓN

Con las ecuaciones 5.3 y 5.4 la plataforma móvil se acerca a la meta a su máxima velocidad, si se encuentra fuera del área de *docking* y comienza a reducirse mientras se encuentra en el área de *docking*. Por otra parte el ángulo se corrige en función de seno del ángulo de error, de forma que si el ángulo de error llega a cero, la velocidad angular también lo será.

Una vez que se tiene cómo llegar a la meta, lo único que queda es planear la trayectoria. Como se mencionó anteriormente, la técnica descrita solo lleva a la plataforma móvil de un punto inicial a uno final, de forma que si se quiere llegar a este punto final con una orientación deseada, se tiene que realizar una planeación.

En el capítulo anterior se estipuló que el punto de referencia es el punto de operación p del efector final. Lo que quiere decir que este punto debe de trazar la trayectoria deseada, por lo tanto, este punto es la guía del Manipulador Móvil. El brazo manipulador por ser actuado mediante servomotores de aeromodelismo, su movimiento es restringido. Esto lleva a que la plataforma móvil, no pueda llegar a la meta en cualquier orientación, sino, en una orientación que se encuentre dentro del limitado rango de movimiento del brazo manipulador. De aquí que sea necesario planear la trayectoria, para esto se tienen que considerar lo siguiente:

- El Manipulador Móvil se desplaza en su entorno con el fin de llegar a un punto meta, en el que se ubica el objeto a manipular.
- El objeto a manipular tiene una posición en el espacio y una orientación.

El hecho de que el objeto a manipular tenga una orientación, implica que no puede ser tomado desde cualquier posición. Por lo tanto, el manipulador móvil, debe de orientarse lo más posible al ángulo de orientación del objeto en el plano $x - y$, permitiendo que el desplazamiento angular del brazo manipulador se encuentre en su rango. Por lo mencionado anteriormente, se tiene que ubicar una sub meta la cual será un *pre - docking* (pD) de la plataforma móvil, este hecho se ilustra en la Figura V – 2.

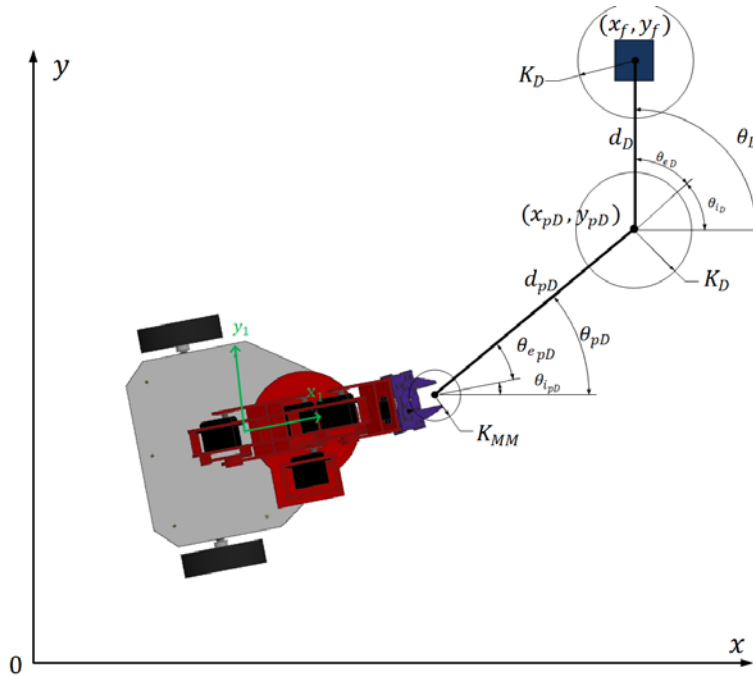


Figura V – 2. Planeación de trayectoria.

Entonces, para calcular el punto pD teniendo como único dato la postura del objeto a manipular, hay que desplazar el punto una distancia d_D , mediante la siguiente ecuación:

$$x_{pD} = x_f - d_D \cos \theta_D \quad (5.5)$$

$$y_{pD} = y_f - d_D \sin \theta_D \quad (5.6)$$

Con las ecuaciones 5.5 y 5.6 se encuentran las coordenadas del punto pD , el cual está a una distancia d_D del punto final y a la misma orientación que posee el objeto a manipular, representado por el punto final. En la figura anterior se puede apreciar dos radios ubicados en los puntos final y pD , estos radios son los radios que en la ecuación 5.3 se expresan como K_r , y son los encargados de reducir la velocidad de la plataforma móvil en cuanto ésta entre en el área descrita por este radio. Se aprecia que son de una misma longitud K_D , por lo tanto para que la plataforma detenga su movimiento debe de incidir el radio K_D con el radio K_{MM} , que se ubica en el punto de referencia del manipulador móvil (punto de operación p), teniendo que:

$$v = 0, \omega = 0 \quad \text{si} \quad \begin{cases} d_{pD} \leq K_{MMpD} \\ d_D \leq K_{MMD} \end{cases} \quad (5.7)$$

En la expresión 5.7, se indica que si la distancia del punto de operación p al punto de pre-docking o de docking, según sea al caso, es menor al radio K_{MM} se detiene el robot. También se nota que hay dos diferentes radios K_{MM} , uno para cada punto. Lo anterior se debe a que para el primer punto (pD), este radio tiene que ser muy pequeño, lo suficiente para que la plataforma no pierda su rumbo

original, pues este es el punto donde la plataforma “da vuelta” para coincidir con la orientación del objeto a manipular. Finalmente, el radio K_{MM_D} indica que la plataforma ya se encuentra orientándose para recoger el objeto y este radio es de mayor longitud al anterior, pues el robot tiene que detenerse a una distancia suficiente, para que el manipulador recoja el objeto a manipular.

En resumen, al algoritmo de campos potenciales, primero se realiza con un punto de *pre – docking*, una vez que se haya llegado a este punto se realiza de nuevo la técnica, con la diferencia de que se realiza con el punto final, deteniéndose cuando la distancia sea la requerido y entonces el manipulador “recoja” su objetivo.

V. 2. OPERACIÓN *PICK AND PLACE*.

El movimiento realizado por un brazo manipulador debe ser, por regla, tan suave como sea posible, esto quiere decir que cambios abruptos en posición, velocidad y aceleración deben evitarse. Movimientos abruptos requieren cantidades grandes de energía para realizarlos. Para esto, se tiene que planear una técnica que permita trazar trayectorias suaves y sin cambios abruptos. Existen típicamente dos técnicas para planear trayectorias, estas son:

- Operaciones *pick and place*.
- Caminos continuos.

En este trabajo la operación *pick and place* se utiliza para planear el movimiento del brazo manipulador, para mayor detalle y información sobre la segunda técnica consultar las referencias [17], [18] y [19].

V. 2. 1. PERFIL DE TRAYECTORIA

En la operación *pick and place (P&P)*, se requiere que el brazo manipulador lleve a una pieza de trabajo u objeto de una posición inicial, con cierta orientación y posición referidas a un marco coordinado, a una posición final. En principio existen dos formas de planear la trayectoria: en el espacio cartesiano o global y en el espacio de juntas. Para el primer caso, se requiere conocer los puntos físico por los cuales va a pasar el órgano terminal del brazo manipulador y el tiempo de recorrido, para ir de un punto de inicio a un punto final. Las posiciones iniciales y finales para la operación *P&P* se describen en el espacio cartesiano, mientras que el movimiento del robot se encuentra en el espacio de juntas. La planeación de la trayectoria del manipulador está compuesta por un lugar geométrico y un perfil de trayectoria.

Por lo tanto, la planeación de la operación *P&P* se realiza para el punto de operación p . Entonces, la posición inicial del punto de operación p está definida por el punto p_i , mientras que su posición final está definida por el vector p_f , éstas

referenciadas al marco coordinado global o espacio cartesiano. Teniendo así que, \dot{p}_i y \ddot{p}_i sean las velocidades y aceleraciones iniciales del punto p , de igual forma que \dot{p}_f y \ddot{p}_f denotan las velocidades y aceleraciones finales del punto p . Por otro lado se asume que para la posición inicial $t = 0$. Posteriormente si la operación se considera que transcurre en un tiempo T , el tiempo para la posición final es $t = T$. Finalmente, se tiene el conjunto de condiciones necesarias para que el movimiento sea suave y continuo desde la posición inicial a la posición final:

$$p(0) = p_i \quad \dot{p}(0) = 0 \quad \ddot{p}(0) = 0 \quad (5.8)$$

$$p(T) = p_f \quad \dot{p}(T) = 0 \quad \ddot{p}(T) = 0 \quad (5.9)$$

Para cumplir con las condiciones anteriormente expuestas una interpolación de polinomios de bajo orden no la satisface, de forma que se utiliza un polinomio de un orden mayor, con la finalidad de cumplirlas. En este trabajo se propone un polinomio de quinto orden para planear la trayectoria, también conocido como perfil de trayectoria. El polinomio se representa en la siguiente expresión:

$$s(\tau) = a\tau^5 + b\tau^4 + c\tau^3 + d\tau^2 + e\tau + f \quad (5.10)$$

Donde el polinomio y su variable poseen el intervalo que se muestra abajo:

$$0 \leq s \leq 1 \quad , \quad 0 \leq \tau \leq 1 \quad (5.11)$$

Por lo tanto la variable independiente del polinomio expresado en 5.10, está definida por la siguiente relación:

$$\tau = \frac{t}{T} \quad (5.12)$$

La forma de describir el lugar geométrico de la posición inicial a la final del punto de operación p , se determina por:

$$p(t) = p_i + (p_f - p_i) s(\tau) \quad (5.13)$$

$$\dot{p}(t) = \frac{1}{T} (p_f - p_i) s'(\tau) \quad (5.14)$$

$$\ddot{p}(t) = \frac{1}{T^2} (p_f - p_i) s''(\tau) \quad (5.15)$$

Entonces los coeficientes que aparecen en la ecuación 5.10, se calculan considerando las condiciones que se expresan en 5.8 y 5.9 y las relaciones mostradas en 5.13 – 15, encontrando las condiciones:

$$s(0) = 0, \quad s'(0) = 0, \quad s''(0) = 0, \quad s(1) = 0, \quad s'(1) = 0, \quad s''(1) = 0 \quad (5.16)$$

Por consecuencia, las derivadas de la ecuación 5.10 se muestran abajo, estas son importante obtenerlas, con el fin de encontrar los coeficientes de $s(\tau)$.

$$s'(\tau) = 5a\tau^4 + 4b\tau^3 + 3c\tau^2 + 2d\tau + e \quad (5.17)$$

$$s''(\tau) = 20a\tau^3 + 12b\tau^2 + 6c\tau + 2d \quad (5.18)$$

Con las condiciones expresadas en 5.16 se sustituyen las expresiones 5.10, 5.17 y 5.18 de las que se encuentra que:

$$f = 0, \quad e = 0, \quad d = 0 \quad (5.19)$$

Además de las expresiones de 5.19, también se forma un sistema de tres ecuaciones con tres incógnitas, que se muestra abajo.

$$a + b + c = 1$$

$$\begin{aligned} 5a + 4b + 3c &= 0 \\ 20a + 12b + 6c &= 0 \end{aligned} \quad (5.20)$$

Resolviendo el sistema representado en 5.20, se obtienen los coeficientes del polinomio:

$$a = 6, \quad b = -15, \quad c = 10$$

Con lo anterior, se encuentra el polinomio de quinto orden buscado, el cual se muestra en la siguiente expresión:

$$s(\tau) = 6\tau^5 - 15\tau^4 + 10\tau^3 \quad (5.21)$$

De acuerdo a 5.11, el rango del polinomio va de $[0, 1]$, como se muestra en la Figura V – 3. En la figura se aprecia que se cumple con la condición de ser un trazo suave y uniforme. Por el hecho mencionado con anterioridad las derivadas del polinomio 5.21 no se muestran. Los servomotores encargados de brindar movimiento a las juntas del manipulador, sólo se les puede proporcionar posiciones y no velocidad como es ampliamente utilizado. Por lo tanto, el perfil de trayectoria se emplea con la finalidad de cumplir con las condiciones para el polinomio y sus derivadas.

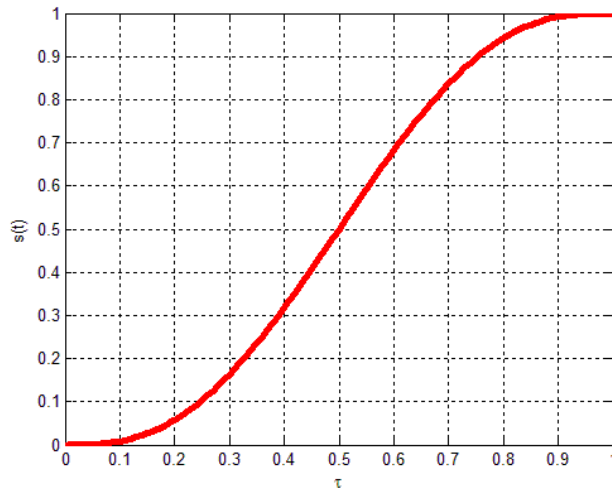


Figura V – 3. Polinomio de interpolación 3-4-5

V. 2. 2. DETERMINACIÓN DEL LUGAR GEOMÉTRICO

Con el proceso anterior expuesto, se puede planear la trayectoria que describirá el efector final del manipulador, solo queda determinar qué puntos son los que la describen.

El objeto a manipular, se considera que se encuentra en algún punto del espacio por el cual se desplaza el manipulador móvil y como se menciona en la sección anterior este se detendrá una cierta distancia (5.7) para manipular. Esto lleva a que la operación *P&P* parte de una posición inicial del manipulador y que el origen del manipulador se encuentra orientado de acuerdo a la postura que, tome la plataforma al llegar al *docking*. Para el primer problema es necesario indicar la

forma en que el manipulador “llegará” al objeto, se tiene que considerar lo siguiente:

- El brazo manipulador empieza su movimiento con una postura x_i, y_i, z_i la cual se considera el inicio y final del movimiento planeado a realizar. Se miden a lo largo del marco del origen del manipulador.
- El objeto tiene como coordenadas x_f, y_f, z_f éstas tienen que estar referidas al marco coordenado origen del manipulador.

La operación como el nombre lo dice es *pick* (recoger) el objeto y *place* (colocar) el objeto en su determinado destino, por lo que el manipulador debe de recoger el objeto. Una vez que lo recoge se mantiene en una posición de forma que se pueda desplazar el robot y colocar el objeto a dónde le corresponde. El anterior párrafo se puede expresar de la siguiente manera:

- Primera trayectoria: de la posición inicial del manipulador x_i, y_i, z_i a un punto en el mismo plano $x - y$, lo que lleva a que, el segundo punto para la trayectoria está dado por x_f, y_f, z_i .
- Segunda trayectoria: partiendo del final de la primera trayectoria (x_f, y_f, z_i) , se dirige el manipulador a un punto en el mismo plano z , por lo que el segundo punto para la trayectoria corresponde a x_f, y_f, z_f . Y al final de ésta, la pinza se cierra, sujetando al objeto, cumpliendo con la operación *pick*.
- Tercera trayectoria: partiendo del punto final de la trayectoria anterior y teniendo sujetado el objeto, se regresa a la posición inicial de forma inversa a como se llegó, es decir, que esta trayectoria se dirige de x_f, y_f, z_f a x_f, y_f, z_i .
- Cuarta trayectoria: finalmente, del punto x_f, y_f, z_i se dirige el manipulador a la posición inicial con la que se parte. Completando con la primera parte de la operación.
- El manipulador móvil se desplaza y llega al destino del objeto realizando nuevamente los cuatro pasos anteriores, sólo que con su correspondiente posición final teniendo sujetado el objeto. Con la diferencia de que al final de la cuarta trayectoria suelta el objeto. En este paso se termina la operación *P&P*, con la colocación (*place*) del objeto en su destino.

El anterior proceso describe el lugar geométrico que tendrá la trayectoria. Sin embargo, en la ecuación 5.13, se muestra el lugar geométrico en base al punto de operación p , si este punto consiste de las coordenadas x, y, z entonces para describir la trayectoria en las componentes del punto de operación se tiene:

$$p(t) = \begin{cases} x_i + (x_f - x_i)s(\tau) \\ y_i + (y_f - y_i)s(\tau) \\ z_i + (z_f - z_i)s(\tau) \end{cases} \quad (5.22)$$

De esta forma, se describe la trayectoria planeada mediante el lugar geométrico de una recta por dos puntos, garantizando que se llegue al punto final al mismo tiempo en sus tres componentes.

Finalmente, el segundo problema que se menciona más arriba es referente a los sistemas coordenadas en los que se encuentran expresados los sistemas. El punto final (x_f, y_f, z_f) determina dos cosas: el punto al cual el manipulador móvil se desplaza y la coordenada del objeto, que se planea manipular. Por lo anterior se tiene que para ambos casos es el mismo punto pero el sistema de referencia es diferente. Siendo para el manipulador móvil expresado en coordenadas cartesianas globales y para el manipulador, el punto está referido al marco coordenado de su origen, debido a la forma de calcular su cinemática. El planteamiento anterior determina, que se tiene que transformar las coordenadas de un sistema al otro. Por las características de la tarea a realizar y el sistema con el que obtiene la ubicación de los componentes, las coordenadas de todos se encuentran referenciadas a un marco global, esto quiere decir, que las coordenadas globales del objeto se tiene que transformar a coordenadas del marco origen del manipulador. Así el manipulador pueda realizar la operación *P&P* del objeto.

Para resolver el presente problema las coordenadas del punto final se encuentran rotadas, el ángulo θ de orientación del manipulador móvil y trasladado del centroide de la plataforma al origen del manipulador. Lo anterior se aprecia en la Figura V – 4, se aprecia que es el mismo punto en coordenadas globales y en coordenadas para el brazo del manipulador. Recordando lo que se expone en el capítulo IV, sobre el acoplamiento de ambas partes del manipulador móvil, donde el punto de referencia se cambia de marco de referencia. Se aplica para este problema la misma idea, por lo que las ecuaciones 4.61 y 4.62 para este problema son las siguientes:

$$\begin{bmatrix} x_r \\ y_r \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x_o + x_m \\ y_m \end{bmatrix} \quad (5.23)$$

Donde se muestra que existe una rotación bidimensional en un ángulo θ , producida por el ángulo del robot y una traslación por la ubicación de ambos marcos coordenados. La expresión de arriba, no otorga las coordenadas en el marco origen del manipulador, sino es función de ella, de manera que despejando los términos x_m, y_m , se encuentra la relación inversa obteniendo:

$$\begin{aligned} x_m &= (x_r - x) \cos \theta + (y_r - y) - x_o \\ y_m &= (y_r - y) \cos \theta - (x_r - x) \sin \theta \end{aligned} \quad (5.24)$$

Que es la relación inversa desarrollada de 5.23. Con esta expresión se encuentran las componentes x, y del punto final para realizar la operación *P&P*, sólo falta la componente z del punto en cuestión, debido a que el modelo de la plataforma móvil aplica cuando el sistema se encuentra en el plano $x - y$, la componente z es la misma y no hace falta convertir.

En las expresiones 5.23 y 5.24 se encuentra un término x_o , el cual se define por la figura, como la distancia entre el origen del marco $x_1 - y_1$ y el marco $x_m - y_m$, en la componente x . Debido a que no hay componente en y no aparece un término que lo represente. Por la configuración del MM diseñado en el capítulo II, esta longitud tiene como valor numérico:

$$x_o = 8.67 \text{ cm}$$

La posición inicial del manipulador es conocida por razones que se explican más adelante y se encuentran referidas al marco origen del manipulador. Por lo que, teniendo ambos puntos en el mismo marco de referencia se traza el lugar geométrico, descrito en ese mismo marco, así el manipulador realizará exitosamente la operación *P&P*.

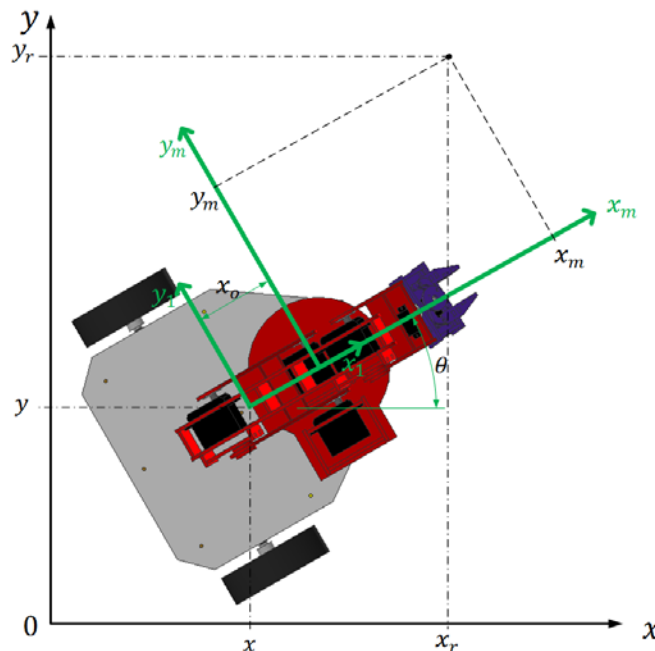


Figura V – 4. Transformación de coordenadas globales a coordenadas del manipulador.

V. 2. 3. ORIENTACIÓN DEL EFECTOR FINAL.

El punto p del efector final, describe la trayectoria que el brazo manipulador debe trazar para recoger el objeto. Como arriba se menciona la segunda trayectoria para cumplir con la primera parte de la operación *P&P* (recoger) va del punto (x_f, y_f, z_i) al punto (x_f, y_f, z_f) . Sin embargo, para recoger el objeto, se propone aumentar el ángulo respecto a la horizontal, desde una posición paralela al piso hasta una posición perpendicular al mismo, para que el objeto sea sujetado por la parte superior de su cuerpo, otro factor es que compensa el hecho de no existir uno de los tres GDL de orientación de la muñeca. Por lo tanto, la posición angular de la última junta del brazo manipulador debe coincidir con el ángulo de orientación del objeto a recoger o el destino a colocar, según sea el caso, así, recoger el objeto de forma correcta. De acuerdo a lo anterior descrito se tiene la

siguiente figura. En la figura se muestra la solución propuesta, para obtener el ángulo que permite al *gripper* coincidir con la orientación del objeto.

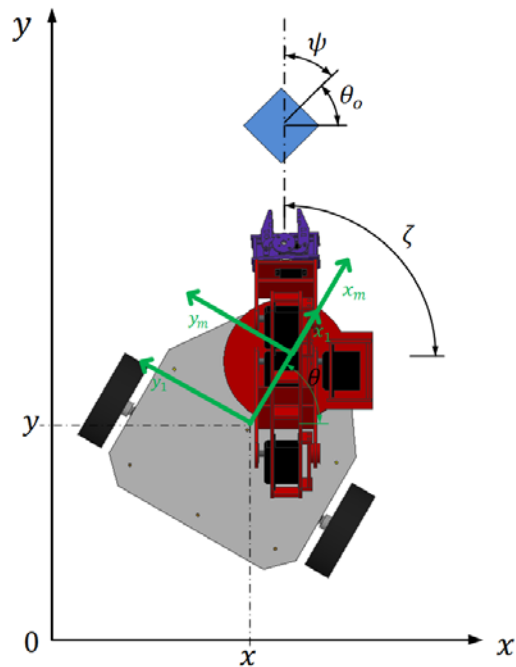


Figura V – 5. Determinación del ángulo de la última junta del brazo manipulador.

Recordando la ecuación 4.38, ψ corresponde al ángulo de la junta cinco. De la figura de arriba se encuentra que el ángulo ψ se encuentra mediante la siguiente expresión:

$$\psi = \zeta - \theta_o \quad (5.25)$$

Donde θ_o , representa el ángulo de orientación del objeto a manipular y ζ , representa el ángulo entre el origen del manipulador y la posición del objeto a manipular en coordenadas globales, se encuentra mediante:

$$\zeta = \text{atan} \frac{y_o - y_{origen_m}}{x_o - x_{origen_m}} \quad (5.26)$$

Donde el punto (x_o, y_o) , se refiere a las componentes del objeto a manipular. Las coordenadas del origen del brazo manipulador, en el marco global, se encuentran mediante la siguiente expresión:

$$\begin{aligned} x_{origen_m} &= x + x_0 \cos \theta \\ y_{origen_m} &= y + x_0 \sin \theta \end{aligned} \quad (5.27)$$

De esta forma se obtiene el ángulo de la quinta junta del brazo manipulador. Sin embargo, los servomotores tienen un rango de movimiento limitado, por lo que hay que acotar este resultado. De forma que, si el ángulo $\theta_5 = \psi$ es negativo se le suma 180° , con lo que el *gripper* puede orientarse correctamente y recoger el objeto adecuadamente.

V. 3. ALGORITMO DE COORDINACIÓN.

Para describir el algoritmo de coordinación heurístico que se propone, se necesita describir primeramente la configuración preferida del manipulador. Este concepto represente la estrategia para coordinar como lo muestra Yamamoto [6]. Al definir este concepto y aplicado a la correspondiente parte del robot, se determina el punto inicial para la operación *P&P*, también se aclara la planeación de la plataforma. Es necesario recordar el acoplamiento que se describió en el capítulo IV, siendo éste, el dato que une a ambas partes y determina la coordinación de movimientos.

V. 3. 1. CONFIGURACIÓN PREFERIDA

Para encontrar la configuración preferida, se usa el concepto de índice de manipulabilidad y la configuración preferida es la que maximiza este índice. El índice de manipulabilidad es considerado como: la distancia entre la configuración del manipulador y las configuraciones singulares, en donde, este índice se vuelve cero. Cerca de las configuraciones singulares al punto de operación del brazo manipulador, le “cuesta trabajo” cambiar la dirección de su movimiento. Principal motivo de utilizar este índice, es mantener lejos de las singularidades al manipulador, cuando éste se encuentre transportando al objeto. Con el fin de que, si en su trayecto el manipulador encuentra perturbaciones, éstas no interfieran con el cumplimiento de la tarea. Debido a la gran capacidad de procesamiento para encontrar este índice, no se puede realizar cuando el robot está ejecutando la tarea. De ser posible, se evaluaría cada punto al que el manipulador se mueva y se encontraría una trayectoria óptima. Por lo tanto, se realiza fuera de línea y se utiliza el resultado para coordinar los movimientos. El índice de manipulabilidad se define como [6]:

$$w = \sqrt{\det(J(\theta)J^T(\theta))} \quad (5.28)$$

Donde w es el índice de manipulabilidad y J representa el Jacobiano del manipulador, a su vez J^T representa la traspuesta del Jacobiano. Con lo anterior se demuestra, que es necesario calcular el Jacobiano del manipulador. Para calcular el índice de manipulabilidad, se consideran solamente las tres primeras juntas, despreciando a las últimas dos juntas que definen a la muñeca. Se realiza con la finalidad de que sea fácil de calcular, sin perder objetividad del método.

De acuerdo a la ecuación 4.8, donde se describe la posición del punto p en función de los cinco ángulos de las cinco juntas, hay que modificar a manera de que se describa el punto C con las tres primeras juntas. Entonces, la ecuación 4.8 se vuelve:

$$a_1 + Q_1(a_2 + Q_2a_3 + Q_2Q_3a_4) = C \quad (5.29)$$

Con la ecuación de arriba, hay que sustituirle los valores que se encuentran en la Tabla IV – 1, los que están definidos como los parámetros de D – H. Por lo tanto, el punto C se define como:

$$C = \begin{bmatrix} 15 \cos \theta_1 \sin \theta_2 \sin \theta_3 - 15 \cos \theta_1 \cos \theta_2 - 15 \cos \theta_1 \cos \theta_2 \cos \theta_3 \\ 15 \sin \theta_1 \sin \theta_2 \sin \theta_3 - 15 \cos \theta_2 \sin \theta_1 - 15 \cos \theta_2 \cos \theta_3 \sin \theta_1 \\ 15 \sin \theta_2 + 15 \cos \theta_2 \sin \theta_3 + 15 \cos \theta_3 \sin \theta_2 + 9.3 \end{bmatrix} \quad (5.30)$$

Donde, la ecuación 5.30 y todas las posteriores vienen dadas en centímetros. De esta forma, de la ecuación anterior se calcula el Jacobiano, que representa las derivadas parciales de primer orden de cada uno de los componentes del vector C a través de los ángulos de las juntas $\theta_1, \theta_2, \theta_3$. Obteniendo la siguiente expresión:

$$J(\theta) = \begin{bmatrix} 15 \sin \theta_1 (\cos(\theta_2 + \theta_3) + \cos \theta_2) & 15 \cos \theta_1 (\sin(\theta_2 + \theta_3) + \sin \theta_2) & 15 \cos \theta_1 \sin(\theta_2 + \theta_3) \\ -15 \cos \theta_1 (\cos(\theta_2 + \theta_3) + \cos \theta_2) & 15 \sin \theta_1 (\sin(\theta_2 + \theta_3) + \sin \theta_2) & 15 \sin \theta_1 \sin(\theta_2 + \theta_3) \\ 0 & 15 \cos(\theta_2 + \theta_3) + 15 \cos \theta_2 & 15 \cos(\theta_2 + \theta_3) \end{bmatrix} \quad (5.31)$$

Con la anterior expresión se puede calcular el índice de manipulabilidad. Sin embargo, tradicionalmente la forma en la que se conduce un vehículo, en este caso un robot móvil, es “mirando” hacia enfrente similar a un control “look-ahead”. Por lo que se considera que el manipulador se encuentre “viendo hacia adelante”. Para la configuración preferida, lo anterior se hace estableciendo al ángulo de la primera junta como $\theta_1 = 0$. Con lo anterior, este valor se sustituye en el Jacobiano, obteniendo una nueva expresión.

$$J(\theta) = \begin{bmatrix} 0 & 15(\sin(\theta_2 + \theta_3) + \sin \theta_2) & 15 \sin(\theta_2 + \theta_3) \\ -15(\cos(\theta_2 + \theta_3) + \cos \theta_2) & 0 & 0 \\ 0 & 15 \cos(\theta_2 + \theta_3) + 15 \cos \theta_2 & 15 \cos(\theta_2 + \theta_3) \end{bmatrix} \quad (5.32)$$

De acuerdo a lo anterior, el índice de manipulabilidad corresponde a encontrar la distancia más alejada de las singularidades, cuando la primera junta tiene un desplazamiento angular igual a cero, para garantizar que el manipulador móvil se desplace en su entorno hacia adelante. Lo anterior mencionado, se indica en el acoplamiento de ambos sistemas. Una vez teniendo el Jacobiano, se tiene que calcular el índice de manipulabilidad. Esta ecuación trata de un sistema no lineal, la solución no se encuentra fácilmente, además, por tratarse de términos trigonométricos, no se tiene una única solución, sino son múltiples. Por lo tanto, en este trabajo se resuelve este problema mediante una gráfica, a través de la paquetería de MATLAB, se graficó el índice de manipulabilidad mediante mallas. Se define una malla de los dos ángulos θ_2, θ_3 que varíen para un ciclo de 0 a 2π y el índice de manipulabilidad representando el eje z. Encontrando gráficamente, el valor máximo del índice de manipulabilidad, la gráfica obtenida se encuentra en la Figura V – 6. El resultado se muestra a continuación:

$$\theta_2 = 35^\circ \quad (5.33)$$

$$\theta_3 = 71^\circ \quad (5.34)$$

Los ángulos de las juntas que se muestran arriba, corresponden cuando el valor numérico de la junta uno es:

$$\theta_1 = 0^\circ \quad (5.35)$$

Los tres valores anteriores dan como resultado el máximo valor del índice de manipulabilidad, que corresponde a:

$$W_{\max} = 5195.82$$

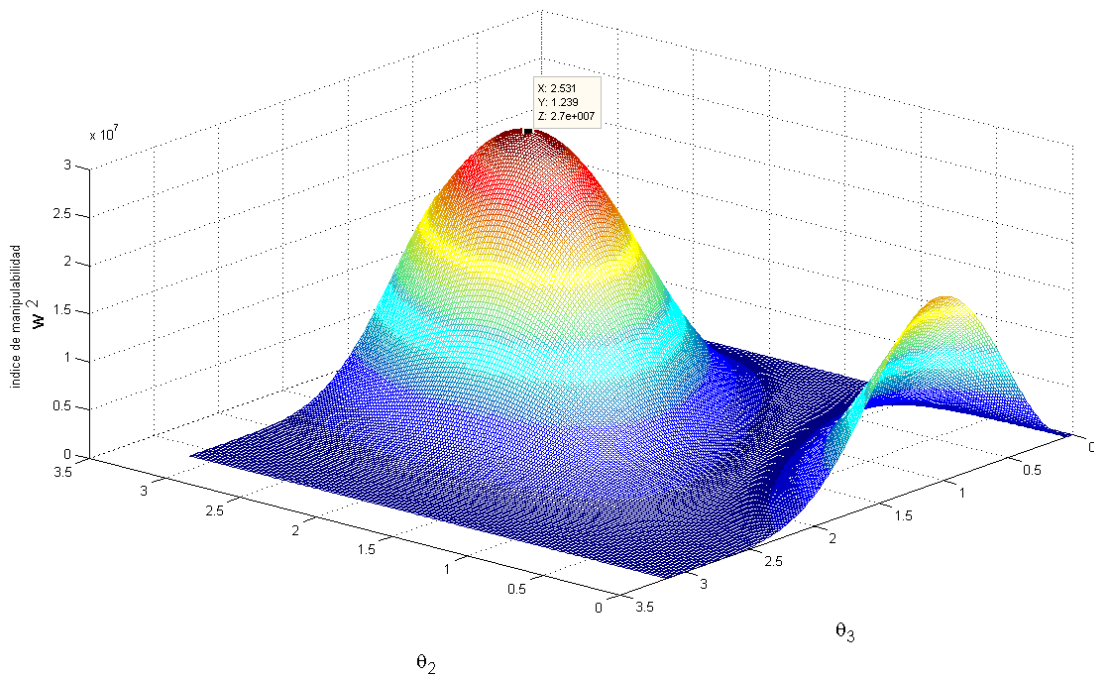


Figura V – 6. Índice de Manipulabilidad.

Los anteriores valores indican el ángulo en que las primeras juntas deben posicionarse, para encontrarse en la configuración preferida. Se nota que falta indicar el valor numérico de las últimas dos juntas, por lo que se tienen que indicar. El valor de la última junta, no influye en la configuración preferida, entonces, uno puede decidir la orientación con la que el objeto se encuentre en la configuración preferida, el valor seleccionado es el siguiente:

$$\theta_5 = 0^\circ \quad (5.36)$$

Sólo queda determinar el valor de la cuarta junta o primera de la muñeca. Este ángulo no está sencillo determinarlo, principalmente porque depende del valor de los ángulos de las juntas anteriores, además existe el parámetro ϕ , dependiente de la orientación del objeto a manipular. Lo que conlleva a seleccionar una orientación para este parámetro, el valor propuesto es $\phi = 0$, con la finalidad de que el objeto esté paralelo al piso en la configuración preferida. En

la Figura IV – 7, se muestra la configuración preferida con el parámetro ϕ asignado y el valor de la última junta, obteniendo para la junta cuatro lo siguiente:

$$\begin{aligned}\theta_4 &= \pi - \theta_3 - \theta_2 \\ \theta_4 &= 74^\circ\end{aligned}\tag{5.37}$$

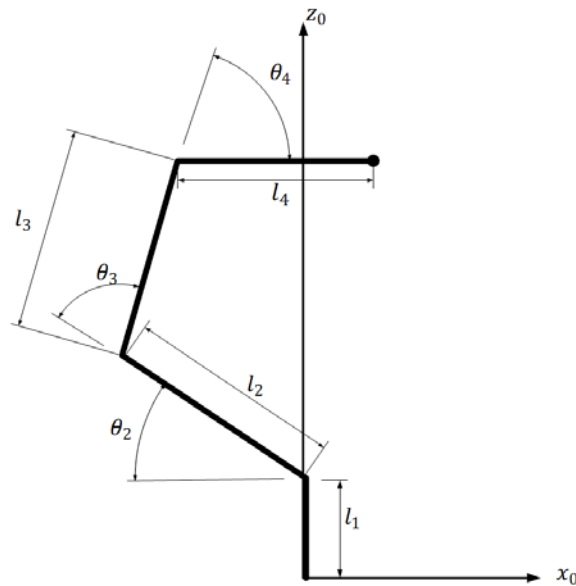


Figura V – 7. Configuración preferida.

V. 3. 2. DEFINICIÓN DEL PUNTO DE REFERENCIA

Con lo que se describe arriba, se tiene completa la configuración preferida, encontrada mediante la maximización del índice de manipulabilidad. Teniendo el valor angular de todas las juntas del brazo manipulador, se puede encontrar el punto inicial para la operación *P&P*, que coincide con el hecho de ser el punto de referencia para la planeación del movimiento de la plataforma móvil. Aplicando la ecuación 4.8, con los ángulos de las juntas en la configuración preferida se obtiene:

$$\begin{aligned}x_i &= 3.54 \text{ cm} \\ y_i &= 0 \text{ cm} \\ z_i &= 31.72 \text{ cm}\end{aligned}\tag{5.38}$$

El acoplamiento descrito en el capítulo IV, definido por las ecuaciones 4.61 y 4.62, indica la estrategia de coordinar. Tomando como punto de referencia la guía para el desplazamiento de la plataforma móvil, este punto, como ya se ha tratado, es el punto de operación *p* del efector final del brazo manipulador, cuando se encuentra en la configuración preferida. Por este hecho se ha encontrado primeramente la configuración preferida y después el punto en las coordenadas origen del manipulador. Lo único faltante es el punto de referencia como lo indican

las ecuaciones 4.61 y 4.62, es decir, falta obtener $({}^1x_r, {}^1y_r)$. La anterior coordenada se encuentra mediante las siguientes expresiones:

$$\begin{aligned} {}^1x_r &= x_o + x_m \\ {}^1y_r &= 0 \end{aligned} \quad (5.39)$$

Siendo esta expresión únicamente válida para la configuración preferida, la cual es utilizada para conducir la plataforma móvil. Sin embargo, la expresión general de la ecuación 5.39, válida para cualquier configuración pero sólo válida para el manipulador móvil diseñado, se muestra a continuación:

$$\begin{aligned} {}^1x_r &= x_o + x_m \\ {}^1y_r &= y_m \end{aligned} \quad (5.40)$$

Las ecuaciones 5.39 y 5.40 se aprecian en la Figura V – 4, donde los términos que conforman las anteriores ecuaciones, se definieron anteriormente. Con la diferencia de que el manipulador se encuentra, en ese momento, en la coordenada indicada.

V. 3. 3. ALGORITMO

Los anteriores párrafos describen la estrategia para tratar a ambas partes como una sola y realizar el movimiento del sistema en su totalidad. Sin embargo, lo anterior sólo determina el movimiento coordinado de ambos sistemas. Por ende falta establecer el algoritmo coordinante de ambos sistemas, este se explica en los siguientes párrafos.

El algoritmo heurístico propuesto consiste básicamente de los siguientes puntos:

1. Detectar manipulador móvil, objetos a manipular y sus correspondientes destinos en coordenadas globales.
2. Seleccionar la secuencia de sub metas a las que tiene que desplazarse el MM, empezando por aquel objeto que se encuentre más cerca del robot.
3. Evaluar en qué parte de la tarea se encuentra el MM y mandar el punto al que debe dirigirse.
4. Llevar a la configuración preferida al brazo manipulador, en cuanto llegue a ella, pasar al siguiente paso.
5. Calcular *pre-docking* y mover la plataforma al mismo.
6. Una vez llegado al *pre-docking*, indicar que cambie el punto al *docking*.
7. Indicar cuando se haya llegado al *docking* e iniciar operación *P&P*, primera parte: recoger.
8. Completada la operación de recoger, indicar que se encuentra nuevamente en la configuración preferida y comenzar movimiento de plataforma, cambiando el punto final al destino del objeto recogido.
9. Calcular *pre-docking* y mover plataforma al *pre-docking*.
10. Una vez llegado al *pre-docking*, dirigirse al *docking*.

11. Cuando se llegue al *docking*, comenzar con la segunda parte de la operación *P&P*: colocar.
12. Colocado el objeto en su destino, regresar a la configuración preferida e indicar que se llegó a ella.
13. Repetir el proceso a partir del paso tres, para el segundo objeto y terminar tarea.

El algoritmo que se describe arriba, demuestra el uso de indicadores (banderas), las cuales permiten pasar y determinar el evento que se realiza en el momento. Revisando constantemente estos indicadores, se coordina la planeación de los movimientos de ambas partes y se coordina la tarea diseñada. A continuación se muestran los indicadores descritos y lo que representan.

Para este trabajo se distinguen dos indicadores:

- IPO: indicador de posición óptima: el nombre lo dice, determina el instante en que se encuentra en la configuración preferida y se le atañe al brazo manipulador, este indicador consiste de tres valores.
 - 0 – Cuando el manipulador se encuentra en la configuración preferida.
 - 1 – Cuando el manipulador se encuentra realizando la operación *Pick & Place*.
 - 2 – Cuando el manipulador termina una parte de la operación *P&P* y regresa a la configuración preferida.
- ILP: Indicador de que llegó al punto: indicador que se le atañe a la plataforma móvil, indica qué evento realiza actualmente la plataforma y consiste de dos valores.
 - 0 – La plataforma móvil se encuentra en movimiento, independientemente del punto al que se dirija (no ha llegado).
 - 1 – La plataforma móvil está estacionada, es decir, ya llegó al punto correspondiente.

Los dos indicadores que determinan qué realiza cada parte del MM, disparan un tercer indicador, que es consecuencia de lo que se realice en el momento, este es:

- IMM: Indicador de evento del manipulador móvil: mediante combinaciones de los anteriores se determina que está realmente realizando la plataforma. Consiste de tres valores.
 - 0 – El manipulador móvil está desplazándose en su entorno.
 - 1 – El manipulador móvil está realizando la primera parte de la operación: *pick* (recoger).
 - 2 – El manipulador móvil está realizando la segunda parte de la operación: *place* (colocar).

El IMM adquiere el valor correspondiente, de acuerdo a la regla establecida para disparar este indicador, esta regla se describe a continuación.

- Si el brazo manipulador se encuentra en la configuración preferida y la plataforma está en movimiento, entonces el manipulador móvil está moviéndose.
 - Si $IPO = 0$ & $ILP = 0$
 - ➡ IMM = 0
- Si el brazo manipulador está en la configuración preferida y la plataforma móvil se detuvo, entonces, el manipulador móvil se encuentra realizando la primera parte de la operación *P&P*.
 - Si $IPO = 0$ & $ILP = 1$
 - ➡ IMM = 1
- Si el brazo manipulador, regresó a la configuración preferida y la plataforma móvil está en movimiento, entonces el manipulador móvil se encuentra moviéndose y se dirige al siguiente punto donde se encuentra el objeto a manipular.
 - Si $IPO = 2$ & $ILP = 1$
 - ➡ IMM = 0
 - ➡ Mandar cambio de punto al que el robot se dirige.
- Si el brazo manipulador se encuentra realizando la operación *P&P* y la plataforma móvil está moviéndose, entonces el manipulador se encuentra en movimiento.
 - Si $IPO = 1$ & $ILP = 1$
 - ➡ IMM = 1
- Cuando el siguiente punto al cual se dirige el MM es numero par (considerando que la numeración empieza con el primer objeto a manipular), entonces, el manipulador móvil realiza la segunda parte de la operación *P&P*.
 - Si $IMM = 1$ y $\text{contador} \% 2 = 0$
 - ➡ IMM = 2

El anterior proceso descrito forma el algoritmo de coordinación heurístico, propuesto en este trabajo, que a su vez corresponde al motivo del mismo. Siguiendo estas reglas, se ejecuta la tarea definida en el capítulo dos de manera coordinada. Sólo queda implementar todo el sistema en una plataforma, que demuestre el funcionamiento del algoritmo propuesto, con el robot diseñado y fabricado en el capítulo tres, lo que aparece en la siguiente sección.

V. 4. DISEÑO DEL EXPERIMENTO EN *SIMULINK*.

En la sección anterior se explica al algoritmo propuesto para realizar la tarea, de ahí se parte para formar con la paquetería de *Simulink* un programa que permita

demostrar el funcionamiento del algoritmo propuesto. Siendo *Simulink* una de las mejores alternativas para realizar simulaciones o experimentos como el descrito a lo largo de este trabajo.

Simulink de *MathWorks*, consiste de un ambiente para simulaciones de dominio múltiple y diseño basado en modelos para sistemas dinámicos y embebidos. Consiste de un editor gráfico interactivo, para armar y manejar diagramas de bloques. Dentro del modelo diseñado se puede crear, configurar y buscar todas las señales, así como parámetros, propiedades y todo el código asociado al modelo introducido. Otros factores importantes son los bloques funcionales de MATLAB, que permiten llevar los algoritmos de MATLAB a *Simulink*, con el fin de implementar sistemas embebidos. Además de que toda la información generada, se tiene acceso mediante MATLAB para analizar y visualizar resultados, personalizando el ambiente del modelo, definiendo señales, parámetros y datos de prueba. La misma plataforma analiza y diagnostica el modelo para asegurar consistencia e identifique los errores. A su vez, MATLAB permite realizar análisis de datos, visualización y cálculos numéricos, además que puede resolver problemas numéricos más rápido que los lenguajes de programación tradicionales.

Estableciendo el software en el que se diseña el experimento, se procede a diseñar el programa, de acuerdo a la descripción del algoritmo explicado en la sección anterior. El programa de *Simulink* se hace por bloques, de aquí en adelante cuándo se indique la existencia de algún bloque se refiere a un componente del programa.

Similar al enfoque de Yingshu en [1], este programa está estructurado en bloques y un conjunto de bloques representa una etapa o un nivel del programa, destacando principalmente las siguientes etapas:

1. Etapa de adquisición: Mediante visión se obtienen las posiciones en el plano del MM, de los objetos a manipular y de los destinos donde se colocan los objetos (se detalla en la siguiente sección).
2. Etapa de decisión o selección: En esta etapa, los dos indicadores ILP e IPO entran para ser procesados y se determina el valor de IMM, así como el punto al que se dirige el manipulador móvil.
3. Etapa de procesamiento: En esta etapa, se procesa toda la información que manda la etapa de selección y a su vez entrega los indicadores ILP e IPO. Lo que quiere decir que en esta etapa se encuentra la planeación de trayectoria, la operación *P&P*, el acoplamiento y los modelos cinemáticos de ambas partes.
4. Etapa de ejecución: En esta etapa, se encuentran los bloques que forman el acondicionamiento de las señales que tienen que ser mandadas al MM, para que cada parte cumpla con su cometido. También se encuentran los

bloques para configurar y mandar los datos (se detalla en la sección seis de este capítulo).

5. Bloques de representación gráfica: en estos bloques entran los datos obtenidos que se mandan al robot y se dibujan en la computadora para visualizar la tarea que está en proceso.

En la Figura V – 8 se muestra un diagrama del flujo de datos, el cual representa lo que se describe arriba, aquí se aprecia que varios sistemas son definidos.

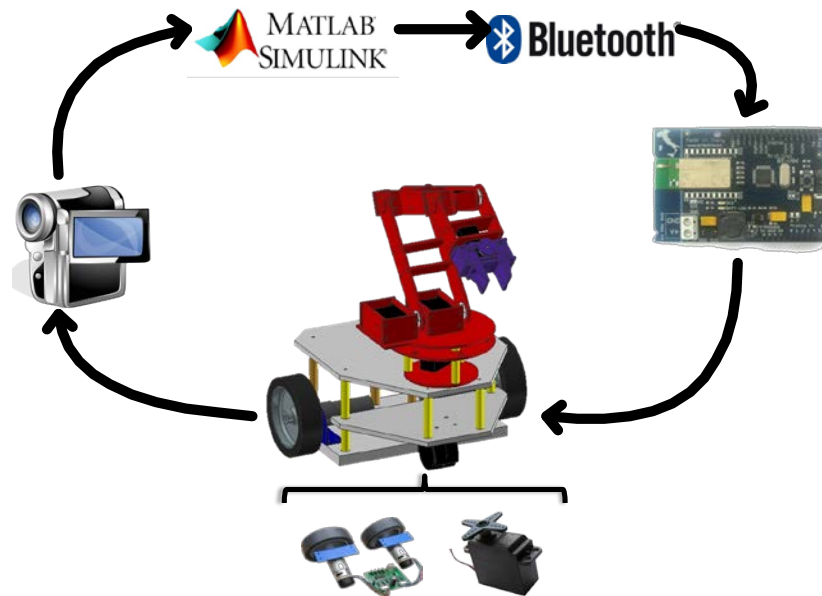


Figura V – 8. Flujo de información.

En la figura, se aprecia que *Simulink* manda los datos por Bluetooth a la tarjeta encargada de proporcionar los movimientos del manipulador móvil. El manipulador móvil consiste de motores de directa y servomotores. La tarjeta debe de mandar la información requerida para que los componentes se muevan y realice la tarea el robot. La posición del manipulador móvil y los objetos a recoger se adquieren mediante el sistema de visión, que a su vez, le proporciona estos datos a *Simulink*. Aquí se procesa la información y se calculan los datos a ser mandados comenzando de nuevo ciclo. Por lo anterior, se tiene que la etapa de adquisición es proporcionada por el sistema de visión, el cual es externo a *Simulink*. En *Simulink* se encuentra la etapa de selección y procesamiento. Finalmente, la etapa de ejecución está formada por bloques de acondicionamiento en *Simulink*, por la comunicación Bluetooth, la tarjeta controladora del MM, que se encuentra externa al programa (lo que indica que cuenta con su propia programación) y los componentes del MM (servomotores, motores CD, tarjeta controladora de motores CD). Lo anterior indica que la etapa de ejecución consiste principalmente de hardware externo al programa, el cual se comunica mediante

visión artificial. El sistema de visión, así como la etapa de ejecución del programa se describirán con mayor detalle en secciones posteriores.

Con todo lo establecido anteriormente, queda describir el flujo de datos interno de *Simulink*, en esta sección se explicarán las etapas selección, procesamiento y los bloques de representación gráfica. La mayoría de los bloques contiene ecuaciones y representaciones trigonométricas o matriciales. Aprovechando que en *Simulink* existen funciones tipo S, las cuales permiten introducir código en el lenguaje de MATLAB con la ventaja de que se ejecutan en *Simulink*, dando pie a tener la información fluyendo y ejecutándose cuando entre información en los bloques de función tipo S. Éstas se utilizan para el diseño del experimento.

Etapa de selección: Esta etapa consiste de un solo bloque, en el cual entran diecinueve señales a procesar, quince de ellas corresponden a la postura (x, y, θ) del robot, de los objetos y de sus destinos, tres por cada elemento. Dos de las señales de entrada representan los indicadores ILP e IPO y dos señales internas: un contador, para saber a cuál de los cuatro puntos se dirige el MM, y una señal que indica el orden al cual debe de dirigirse el MM.

Primeramente, se calcula la distancia de cada objeto a manipular al manipulador móvil. Esto se realiza antes de comenzar el movimiento del robot y determina hacia cuál de ellos debe dirigirse primero. Sabiendo a qué objeto se debe dirigir primero, se obtiene la secuencia de puntos a los que se dirige. La secuencia se acomoda de la siguiente forma: objeto uno, destino del objeto uno, objeto dos, destino del objeto dos.

Conforme los indicadores ILP e IPO entran a este bloque, el algoritmo propuesto determina qué se encuentra desempeñando el manipulador móvil y lo indica a la siguiente etapa mediante el IMM, además de determinar cuándo el cambio de punto se debe realizar. Por lo tanto, además del indicador IMM, de este bloque sale la posición y orientación del punto al que se tiene que dirigir en ese momento el MM. En la Figura V – 9, se muestra el bloque con sus entradas y salidas.

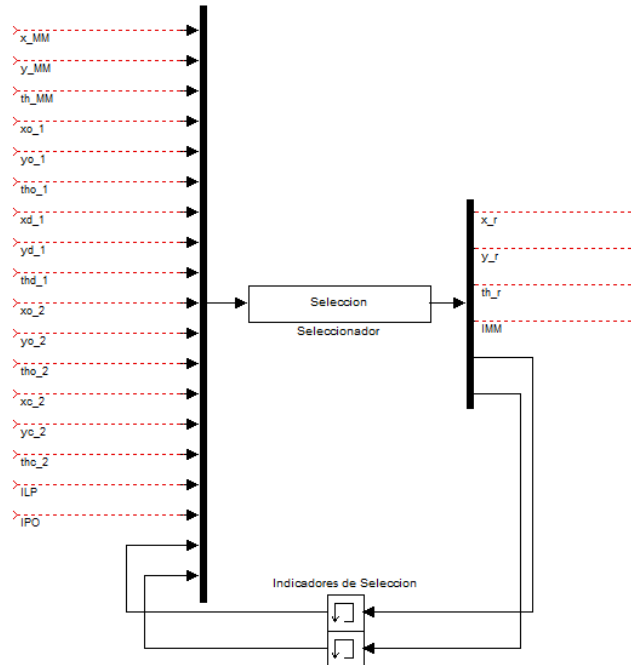


Figura V – 9. Etapa de decisión o selección.

De existir más objetos, se pueden agregar las líneas de código. Así, la etapa de selección determina a qué punto dirigirse, en qué orden y cuándo cambiar de punto.

Etapa de procesamiento: Esta etapa consiste de la operación *P&P*, la cinemática inversa del manipulador para realizar la operación, el acoplamiento, la planeación de la trayectoria de la plataforma móvil y las restricciones cinemáticas de la misma, para trazar la trayectoria.

Primeramente, se encuentra el bloque de la operación *P&P*, el cual, tiene como entradas ocho señales: tres que indican la posición y orientación del objeto a manipular o el destino del mismo; tres que indican la postura de la plataforma móvil; una entrada es indicador IMM; y otra un contador para realizar la trayectoria. A su vez, el bloque tiene ocho salidas: tres son las coordenadas x, y, z a las que el brazo manipulador se tiene que mover, referenciadas al marco origen del brazo, y las que entran al bloque de cinemática inversa; otras dos señales indican el ángulo respecto a la horizontal (ϕ); y el ángulo de orientación de la última junta del brazo; la sexta señal determina si el *gripper* está abierto o cerrado; la séptima señal es el indicador IPO; y finalmente la última señal corresponde a la última entrada de este bloque, como se ve en la Figura V – 10. Básicamente en este bloque, se encuentra el lugar geométrico que define la trayectoria con el perfil establecido en la sección V. 2, como ya se mencionó anteriormente, está formado por dos rectas. Se incluye la transformación de coordenadas, debido a que entra al bloque la posición y orientación del objeto en el marco global. Sin embargo, el indicador IMM le dice a este bloque qué trayectoria debe realizar, si debe cumplir

con la operación *pick* o con la operación *place*. También indica si el manipulador encuentra en la configuración preferida, por consecuencia, se obtiene el indicador IPO que sale a la etapa de selección. Por lo tanto, este bloque determina el movimiento del brazo manipulador correspondiente a lo que señala el indicador IMM. De forma que, las primeras cinco salidas del bloque, entran a la cinemática inversa del manipulador, que corresponde al siguiente bloque.

En el bloque de cinemática inversa, entran cinco señales: las primeras cinco salidas del bloque que determina el movimiento del brazo manipulador (las coordenadas x, y, z en el marco origen del manipulador, el ángulo respecto a la horizontal y el ángulo de orientación de la última junta). Como el nombre lo dice, aquí se encuentran las ecuaciones calculadas para la cinemática inversa del brazo manipulador en el capítulo IV, por ende las salidas de este bloque corresponden al ángulo de cada junta del brazo, como se muestra en la figura V – 10.

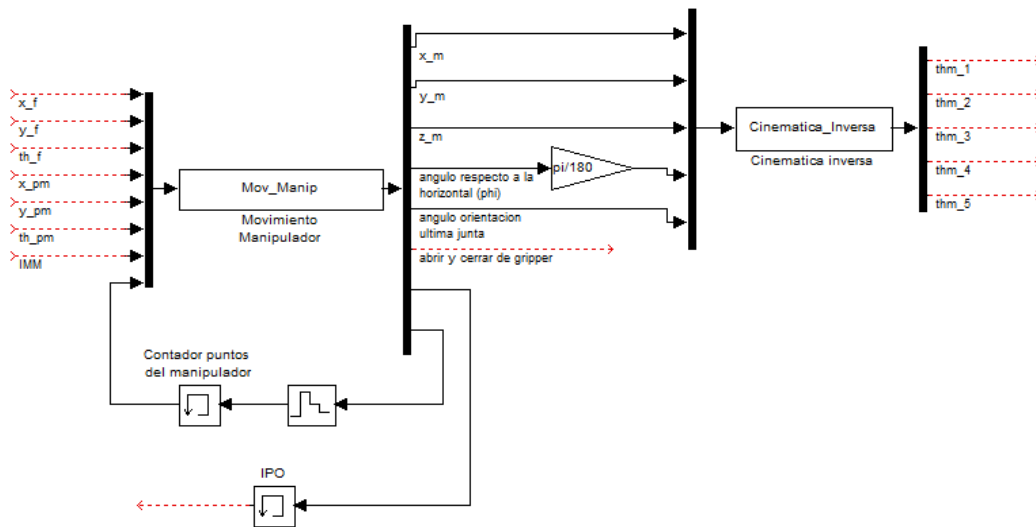


Figura V – 10. Etapa de procesamiento. Bloques de brazo manipulador.

El siguiente bloque, de la etapa de procesamiento, corresponde al acoplamiento mencionado en el capítulo IV. En este bloque entran siete señales: las tres primeras corresponden a la postura de la plataforma móvil; las siguientes tres al ángulo de las primeras tres juntas del brazo manipulador (salidas del bloque se cinemática inversa); y la última al ángulo respecto a la horizontal (ϕ). Este bloque cuenta con tres salidas que corresponden al punto de referencia para la planeación de trayectoria de la plataforma móvil y el ángulo de orientación de la misma. En este bloque, se programan las ecuaciones 4.61, 4.62 y 5.37 y se muestra en la Figura V – 11.

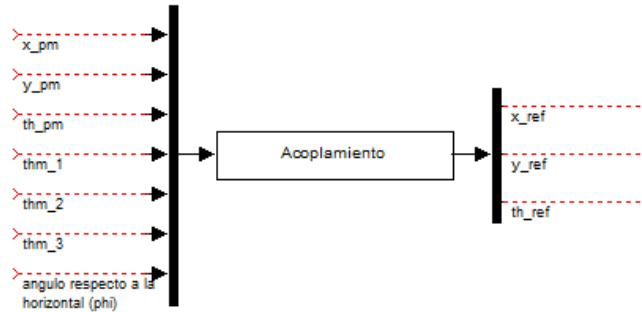


Figura V – 11. Etapa de procesamiento. Bloque de acoplamiento.

Finalmente, los últimos bloques de la etapa de procesamiento, corresponden a la planeación de trayectoria para la plataforma móvil y a las restricciones cinemáticas que permiten el movimiento de la plataforma.

El bloque que le sigue al bloque de acoplamiento es el bloque de “llegando a la meta”, que contiene la planeación de la trayectoria vista en la sección V. 1. Este bloque consiste de ocho entradas: tres corresponden a la salida del bloque de acoplamiento, es decir, la posición del punto de referencia para la plataforma y el ángulo de orientación de la misma; las siguientes tres corresponden a la salida de la etapa de selección y representan la posición final a la que el MM debe llegar, ahí se encuentra su objetivo; entra un contador interno que determina si la plataforma se dirige al *pre-docking* o al *docking*; y la última señal de entrada es el indicador IMM.

En primera instancia, se calcula el *pre-docking* a una distancia d_D del punto meta y al mismo ángulo de orientación. Esta distancia se propone que valga $d_D = 45 \text{ cm}$. De igual forma que en el bloque de movimiento del manipulador, el indicador IMM determina en qué situación se encuentra la plataforma, es decir, si IMM determina que se está realizando la operación *P&P* la plataforma móvil debe estar detenida. De lo contrario, si IMM determina que se está moviendo el robot, la plataforma se mueve. Entonces, el punto de *pre-docking* sólo se calcula si la plataforma se está moviendo. En este bloque, se programan las ecuaciones 5.3 y 5.4, que corresponden a la velocidad lineal de la plataforma móvil y la velocidad angular de la misma. Sin embargo, en este mismo bloque se calcula el *pre-docking*, por lo que el contador interno programado indica si la plataforma ya llegó al punto correspondiente. Cuando llega al *pre-docking* tiene que dirigirse al *docking* y una vez que llega al *docking* se detiene, De acuerdo a la situación será el valor del indicador ILP. Esto lleva a lo que establece la ecuación 5.7, donde se propone lo siguiente:

$$K_{MM_{pD}} = 3 \text{ cm}$$

$$K_{MM_D} = 19 \text{ cm}$$

Estos valores, indican el radio de convergencia para detener la plataforma. Como se mencionó en la sección V.1, uno es menor al otro con el fin de no perder la dirección del punto. Por consecuencia, este bloque tiene como salidas la señal de velocidad lineal y angular para la plataforma, el indicador ILP y el resultado del contador interno, consecuencia de la penúltima entrada, como se aprecia en la Figura V – 12.

Para finalizar las velocidades lineal y angular, así como la orientación de la plataforma móvil, que corresponden a la salida del bloque “llegando a la meta”, entran al bloque que representa las restricciones cinemáticas de la plataforma móvil. En este bloque se programa la ecuación 4.60, aquí, las entradas corresponden a cinco: las tres mencionadas anteriormente; y los parámetros de la plataforma móvil, que de acuerdo al robot diseñado se tiene:

$$d = 4.33 \text{ cm}$$

$$b = 14 \text{ cm}$$

$$r = 5 \text{ cm}$$

Por lo tanto, las salidas de este bloque corresponden a la ecuación 4.60: las velocidades lineales de la componente en x y en y ; la velocidad angular; las velocidades de las llantas izquierda y derecha, como se muestra en la Figura V – 12.

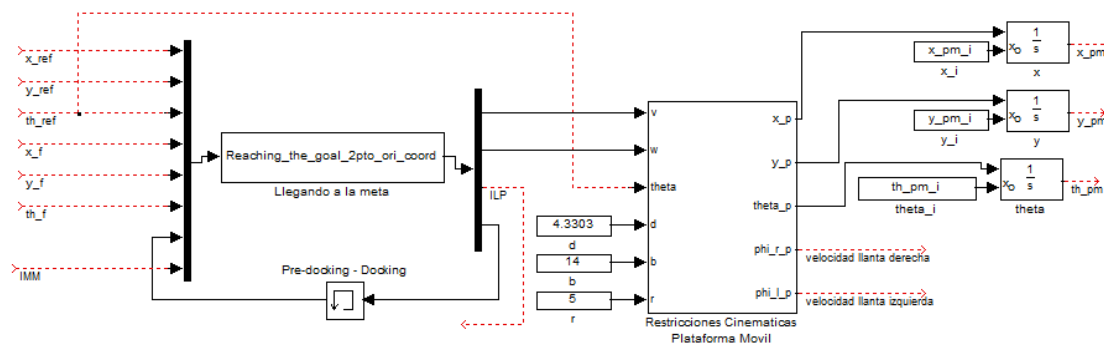


Figura V – 12. Etapa de procesamiento. Bloques de la plataforma móvil.

Bloques de representación gráfica: Consiste de dos bloques, donde uno dibuja el brazo manipulador en coordenadas referenciadas a su origen x, y, z y el otro dibuja la plataforma móvil, junto con los objetivos en el plano $x - y$, en coordenadas globales. En estos bloques entran todas las variables necesarias para dibujar cada caso, de tal forma que durante la ejecución del programa se aprecie la representación gráfica, como se muestra en la Figura V – 13.

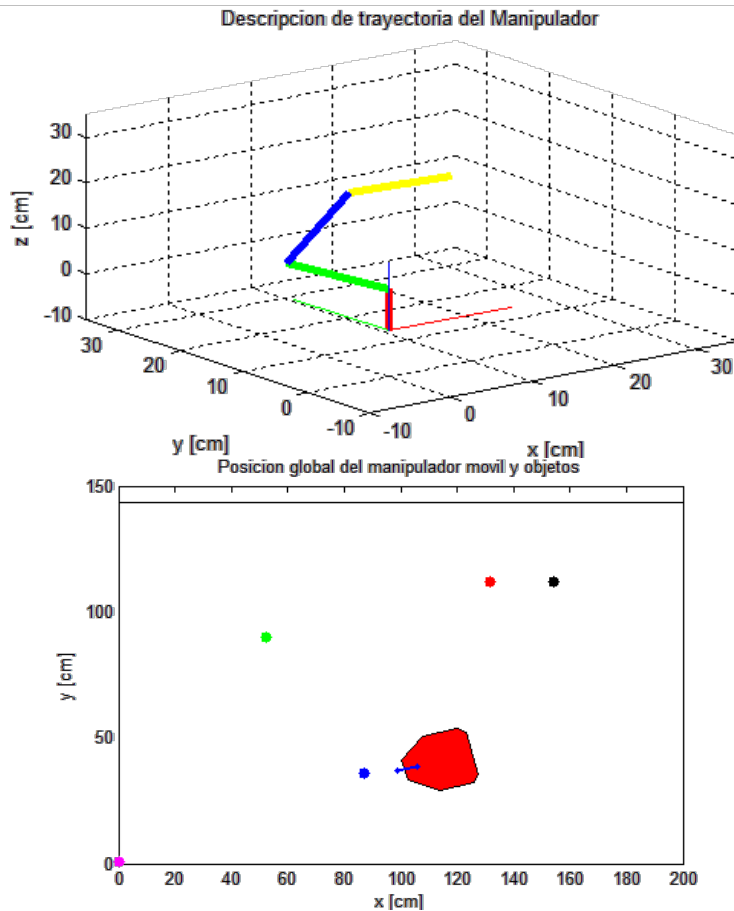


Figura V – 13. Representación gráfica.

V. 5. REACTIVISION.

Mediante el sistema de visión se puede obtener una vista aérea del entorno de trabajo del manipulador móvil. Se requiere la vista aérea debido a que el entorno en el que se desplaza la plataforma es el plano $x - y$, sólo el brazo manipulador es el que trabaja en las tres dimensiones x, y, z . Sin embargo, si se conoce a que coordenada en z se dirige, la vista en el plano $x - y$ es suficiente, representado en coordenadas globales. Por lo tanto, es correcto utilizar la vista aérea del entorno de trabajo donde se desplaza el manipulador móvil, de esta forma se puede apreciar la ubicación del robot, así como las metas a las que tiene que llegar el MM. Entonces, el campo de visión que cubre la cámara representa el límite del espacio de trabajo. Lo que quiere decir que, si el identificador que tiene unido cada objeto sale de este campo, lo perderá. Cada objeto debe tener un identificador para detectarlo en el plano $x - y$.

El software, que permite ubicar cada objeto mediante un identificador en el plano, utilizado en este trabajo es *reactIVision* [22]. Es un software de código libre para el seguimiento robusto de *fiducial* (Figura V - 14) o identificadores que se le

adjuntan al objeto físico a detectar, al igual sirve para aplicaciones multi-touch con la detección de dedos.

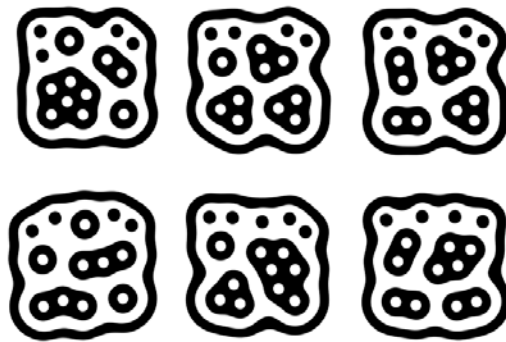


Figura V – 14. Ejemplos de identificadores *Fiducials*.

ReactIVision es un aplicación, la cual manda mensajes TUIO vía UDP puerto 3333 a cualquier aplicación del cliente TUIO disponible. El protocolo TUIO está disponible en varios lenguajes de programación, el cual sirva para comunicar *ReactIVision* con *Simulink*.

Este software funciona de la siguiente manera, primero rastrea los identificadores *Fiducial* capturados en tiempo real por una cámara de video. La imagen fuente es convertida en una imagen blanco y negro con un algoritmo de umbrales adaptable, entonces, esta imagen se segmenta en un árbol de regiones blancas y negras alternadas, formando así una gráfica. En la imagen generada, se busca la secuencia codificada para identificar cada *Fiducial*. Finalmente, el árbol de secuencias encontrado se compara con su diccionario que entrega el numero identificador ID de cada símbolo *Fiducial*. El diseño del *Fiducial* permite un cálculo eficiente del centroide de la figura, así como su orientación. Cuando se identifica totalmente la ubicación, su orientación y el identificador de cada *Fiducial*, la información se envía a través del protocolo TUIO por medio de algún cliente disponible.

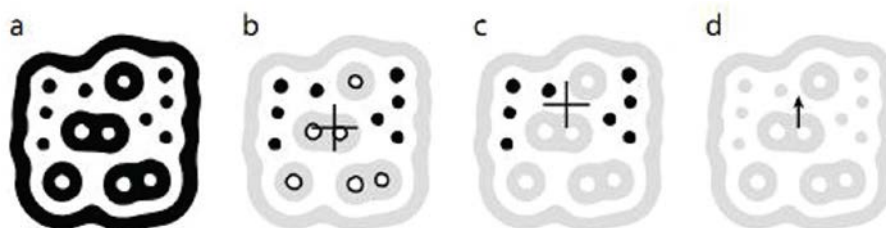


Figura V – 15. Detección de *Fiducials*.

El centroide de la figura se calcula con todos los puntos blancos y negros, mediante una suma ponderada de ellos se encuentra el centroide. Con únicamente los puntos negros se vuelve a calcular el centroide y trazando un vector, dirigido del primero al segundo, se estima la orientación del símbolo *Fiducial*, como se ve en la Figura V – 15.

Establecido lo anterior, puede apreciarse que estos símbolos fiduciales serán los identificadores que permitan saber la ubicación y orientación del MM, de los objetos a recoger y del lugar destino donde se tienen que colocar. Así aprovechando esto, cada uno de los sistemas mencionados tiene unido un símbolo *Fiducial* para su detección en el entorno de trabajo.

Entonces de acuerdo a lo que se menciona en la sección anterior, la detección de los sistemas forma la etapa de adquisición, de la implementación del algoritmo de coordinación propuesto en este trabajo.

Etapa de adquisición: Esta etapa consiste de dos bloques: un bloque donde se adquieren los datos y otro bloque donde se ajustan estos datos. En el primer bloque (Adquisición de datos) no existen entradas, debido a que se adquieren los datos mediante código, pero posee quince salidas que corresponden a las primeras quince entradas de la etapa de decisión o selección. Sin embargo, antes de llegar a la siguiente etapa pasan por el segundo bloque (Ajuste de datos de visión), en este bloque se ajustan los datos que se reciben, debido a ciertas características propias del manipulador móvil y del entorno de trabajo, que se detallarán más adelante, en la Figura V – 16, se muestra el diagrama de bloques que se describe en este párrafo.

En el primer bloque de esta etapa se ejecuta el cliente TUIO, que recibe los datos que *ReactIVision* manda y de esta forma se adquieren los datos. Para el caso de trabajar con MATLAB y *Simulink* el cliente TUIO está en el lenguaje de programación *java* (véase Apéndices), de esta forma se adquieren las posiciones y ubicaciones de los símbolos *Fiducial* que representan cada uno de los sistemas de la prueba.

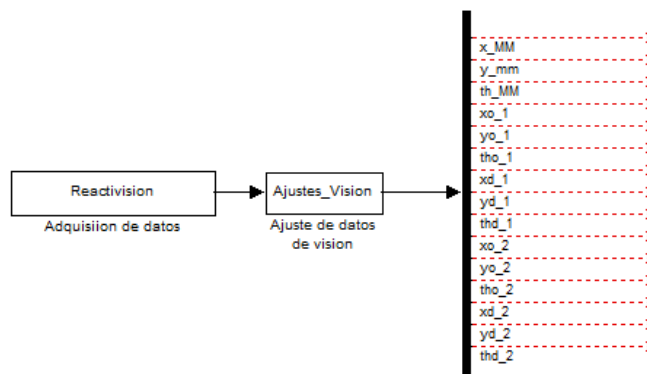


Figura V – 16. Etapa de adquisición.

Sin embargo, como ya se mencionó, se hace un ajuste de estos datos. El primer ajuste se menciona en el siguiente capítulo debido al entorno de trabajo. El segundo ajuste se debe a que el manipulador móvil, principalmente el brazo, obstruye el rayo de visión de la cámara esto, porque la altura del manipulador en su configuración preferida y la plataforma móvil supera los 45 cm. Haciendo que la

detección del símbolo *Fiducial* sea errónea o totalmente nula, provocando que la tarea no se realice con eficacia. Estipulado lo anterior se encuentra que el *Fiducial* no puede estar ubicado en el centro del sistema a considerar, sea los objetos a manipular, sus destinos o el punto P del manipulador móvil. Por lo tanto, la solución propuesta y con la que la respuesta mejora es trasladando el punto de análisis, similar a lo que se hace para calcular el *pre-docking*. Lo que quiere decir que el símbolo *Fiducial* se encontrará una distancia d_o , medida del centroide del *Fiducial* al centro del objeto a manipular o su destino, según sea el caso; con la misma orientación del *Fiducial*, como se ve en la Figura V – 17a. Por otra parte, el *Fiducial* que corresponde al manipulador móvil se encuentra a una distancia d_{MM} medida del punto P al centro del *Fiducial*, como se aprecia en la Figura V – 17b y se define mediante las siguientes expresiones.

$$\begin{aligned}
 x_{o_{i_{real}}} &= x_{o_i} - d_o \cos \theta_{o_i} \\
 y_{o_{i_{real}}} &= y_{o_i} - d_o \sin \theta_{o_i} \\
 x_{d_{i_{real}}} &= x_{d_i} - d_o \cos \theta_{d_i} \\
 y_{d_{i_{real}}} &= y_{d_i} - d_o \sin \theta_{d_i}
 \end{aligned}
 \tag{5.41}$$

$$\begin{aligned}
 x_{MM_{real}} &= x_{MM} + d_{MM} \cos \theta_{MM} \\
 y_{MM_{real}} &= y_{MM} + d_{MM} \sin \theta_{MM}
 \end{aligned}$$

Donde $i = 1, 2$, representa los dos objetos a manipular, el índice o representa objeto y el índice d destino, así como el índice MM se refiere al manipulador móvil.

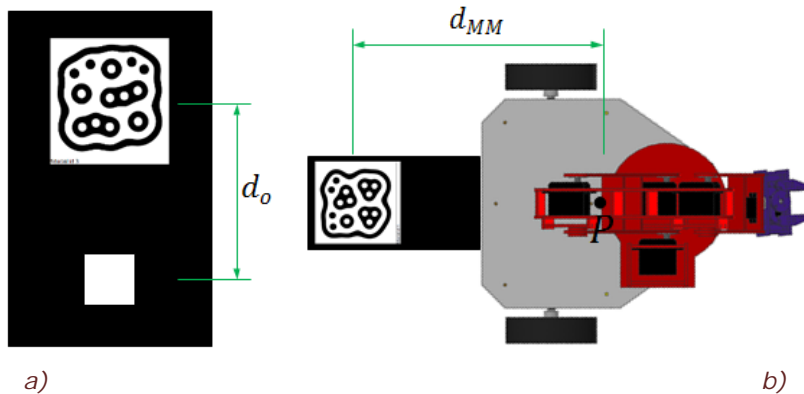


Figura V – 17. Traslado de puntos de detección: a) para objetos y destinos b) para manipulador móvil.

V. 6. COMUNICACIÓN COMPUTADORA-MANIPULADOR MÓVIL.

Debido a que el manipulador móvil es un robot que se desplaza en su entorno, la comunicación alámbrica de éste con la computadora no es una opción. De lo contrario, el mismo robot pasaría por encima de los cables o se desconectarían,

haciendo que una persona se encargue de seguir al robot e impida que esto suceda. Como la cámara con la que se detecta la posición y orientación de los sistemas que componen la tarea, se encuentra montada a una altura tal que permita apreciar todo el entorno de trabajo, la persona que siga al robot estorbaría para su correcta detección. Por lo tanto, la comunicación alámbrica no es una opción, sino, se necesita que sea vía inalámbrica la comunicación entre la computadora (donde está programado el algoritmo coordinador y se realizan todos los cálculos para su ejecución) y el Manipulador Móvil (el robot que ejecuta la tarea diseñada).

Existen varias formas de realizar comunicación inalámbrica, unas más difíciles que las otras. Sin embargo, con el fin de no complicarse en la comunicación (no es objetivo de este trabajo) y considerando la existencia de microcontroladores con módulo de Bluetooth integrado, la comunicación mediante Bluetooth es la que se decide utilizar para implementar el trabajo que se ha estado desarrollando a lo largo de éste. Cumpliendo con el requisito de ser inalámbrica y relativamente sencilla de implementar, por lo que se describe posteriormente, además de poseer una buena velocidad de transferencia de datos. La tecnología inalámbrica Bluetooth es el estándar inalámbrico de banda corta global, que permite la conectividad con una gran cantidad de dispositivos electrónicos. Esta tecnología hoy en día sigue evolucionando; las características más relevantes de esta tecnología es el bajo consumo de energía y su bajo costo, sin perder la robustez que proporciona.

El microcontrolador que se menciona arriba y el que se utiliza para la implementación de la tarea es de la marca ATMEL, montado en una tarjeta de Arduino [23]. El modelo de tarjeta utilizado tiene un módulo de Bluetooth Bluegiga WT11 integrado, con un microcontrolador de montaje superficial modelo ATmega328 (Figura V – 18).



Figura V – 18. Arduino BT

El módulo de Bluetooth Bluegiga WT11 en la tarjeta Arduino BT provee la comunicación con computadoras, teléfonos y otros dispositivos Bluetooth. El módulo WT11 comunica el ATmega328 vía serial (RS232). Viene configurado para

una velocidad de 115200 baudios de comunicación. Este módulo, se tiene que emparejar con la computadora, lo que debería generar un puerto virtual COM. La placa puede ser programada mediante la misma vía inalámbrica. Lo anterior estipula que la comunicación no es directamente Bluetooth, sino por la conversión de un puerto virtual, que se genera por la computadora. Lo que permite que la programación de la placa Arduino BT sea la misma como la de cualquier otra placa de la familia Arduino. Por lo tanto, identificando el puerto virtual COM generado, se programa la placa utilizando comunicación serial tradicional, en la siguiente figura se muestra un esquema de este proceso.

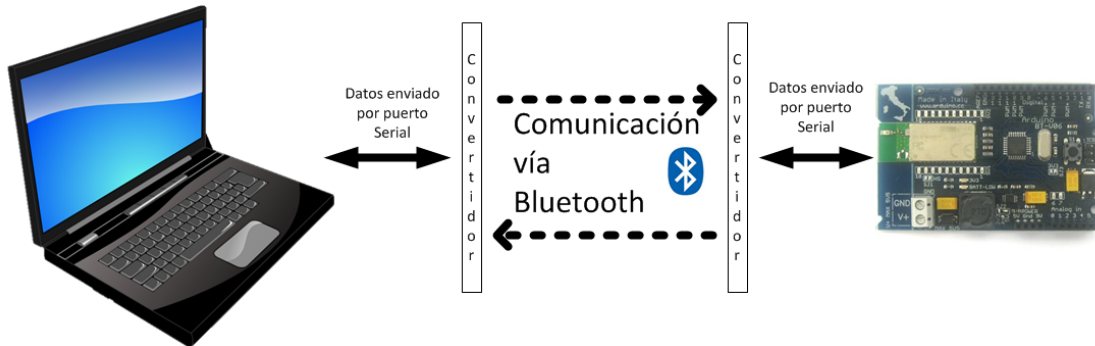


Figura V – 19. Esquema de comunicación computadora – manipulador móvil

Una vez descrito el sistema de comunicación elegido, queda solamente detallar la etapa de ejecución que se estableció anteriormente. Esta etapa está dividida en dos, pues tanto esta sección como la siguiente forman la etapa de ejecución.

Primera parte de etapa de ejecución: En esta primera parte, se plantean los bloques que configuran el puerto serial, para la comunicación entre la computadora y el robot y los bloques que constituyen al acondicionamiento de datos para su posterior envío. Primeramente se tiene que configurar el puerto COM, por el cual se envían los datos. Con el bloque *Serial Configuration*, este puerto ya se estableció, es aquel que se genera al emparejar la computadora con el Arduino BT. En este bloque se configuran los parámetros para la comunicación serial, parámetros como el número de puerto, la velocidad de baudios, los bits de datos, entre otros. De no configurar primeramente el puerto con el que se trabaja, los datos no se enviarían, esto hace surgir que existe un bloque encargado de enviar los datos este bloque es el *Serial Send*. En este bloque entran los datos que son enviados por el puerto que ya se configuró.

Ya que se estableció, que los datos serán enviados puerto serial y la configuración de este, queda determinar qué datos son enviados al Arduino BT. Recordando que el manipulador móvil está formado de un brazo manipulador de 5 GDL donde el movimiento lo realizan servomotores de aeromodelismo y una plataforma móvil en donde se ubican dos llantas motrices, se aprecia que los datos que se tienen que mandar, son aquellos que permitan el movimiento del

robot. Para el manipulador: la posición angular para cada una de sus juntas y la posición angular para abrir y cerrar el *gripper*; y para la plataforma móvil la velocidad de la llanta derecha y de la llanta izquierda. Una vez establecidos los datos a enviar se tienen que hacer acondicionamiento de las señales.

El acondicionamiento de los datos del manipulador son esencialmente tres:

- El primero es un bloque donde se realiza un filtro, es decir, si la posición angular a la cual se tiene que mover cada junta sobrepasa a la del límite determinado por el servomotor, se tiene que mandar una posición de inicio. La cual determina que el manipulador está fuera de rango.
- Después del primer acondicionamiento viene el acondicionamiento de ancho de pulso de los servomotores. Este acondicionamiento se debe principalmente a dos cuestiones:
 1. La tarjeta Arduino tiene una librería que manda la señal a los servomotores que estén conectados, esta señal puede ser determinada por: posición angular, en números enteros, y por ancho de pulso.
 2. La tarjeta Arduino solo recibe Bytes de información por lo que si se manda algún tipo de dato mayor al que puede recibir la información no es procesada correctamente. De ahí que se convierta el tipo de datos por enteros no signados de ocho bits.

Las posiciones angulares que genere la Cinemática Inversa del manipulador obviamente no son enteras en la mayoría de los casos. Por lo que, para mayor precisión tendrán que mandarse los valores con punto flotante, sin embargo debido a la primera restricción se tienen que enviar los datos en ancho de pulso. Entonces, la posición angular generada por la etapa de procesamiento, se tiene que convertir en ancho de pulso para cada servomotor (característica propia de cada tipo de servomotor comercial). Lo que estipula la segunda restricción es que no se pueden enviar datos que superen el Byte y siendo que el ancho de pulso es un número en micro-segundos mayor a 255, lo que se hizo fue separar el valor en ancho de pulso en dos cantidades para cada servomotor.

En resumen, el acondicionamiento para los datos del manipulador son: establecer que no sobrepasen el límite de cada servomotor; que los datos enviados tienen que estar representados en ancho de pulso (para mayor precisión en los movimientos) y divididos en dos cantidades por cada servomotor. Haciendo que se manden diez datos para las juntas del manipulador más un dato que representa el abrir y cerrar del *gripper*.

Los datos divididos que se envían al Arduino BT son procesados y juntados de nuevo así formando el ancho de pulso correcto y escribiéndolo a las juntas del manipulador. La etapa de ejecución se aprecia en la Figura V – 20.

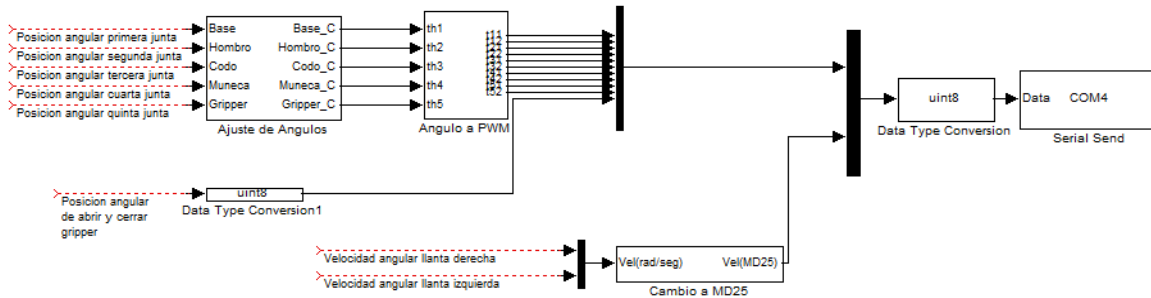


Figura V – 20. Etapa de ejecución.

V. 7. COMUNICACIÓN PLATAFORMA MÓVIL CON ARDUINO BT MEDIANTE PROTOCOLO I²C.

La plataforma móvil está formada por dos llantas, las cuales proporcionan movimiento al robot y así pueda desplazarse en su entorno. Estas llantas están actuadas, es decir, tienen un motor acoplado al eje de giro de las llantas. Entonces, la plataforma móvil utilizada en este trabajo tiene dos llantas actuadas como se mencionó en el capítulo III, los motores CD utilizados son el modelo EMG30. El modelo Cinemático descrito en el capítulo IV, proporciona la velocidad de las llantas que se generan al llegar a un objetivo, lo que quiere decir que, los motores EMG30 de la plataforma móvil, tienen que girar a una velocidad determinada por su modelo. Existe una tarjeta controladora para estos motores, la tarjeta MD25, que se describe a continuación.

La tarjeta MD25 (Figura V – 21) es una tarjeta controladora de doble puente H, desarrollada por *Devantech*, Ltd y está diseñada para controlar dos motores EMG30. Sus principales características son:

- Lectura de *encoders* para determinar el desplazamiento de los motores y el sentido de giro de estos.
- La tarjeta permite conocer el consumo de corriente de cada motor.
- Se alimenta con un voltaje único de +12 VDC.
- Tiene integrado un regulador de +5VDC a 300mA para alimentar circuitos externos (Arduino BT).
- Es capaz de suministrar hasta 3A a cada motor.
- Cuenta con la posibilidad de comunicarse a ella mediante protocolo serial y protocolo I²C,

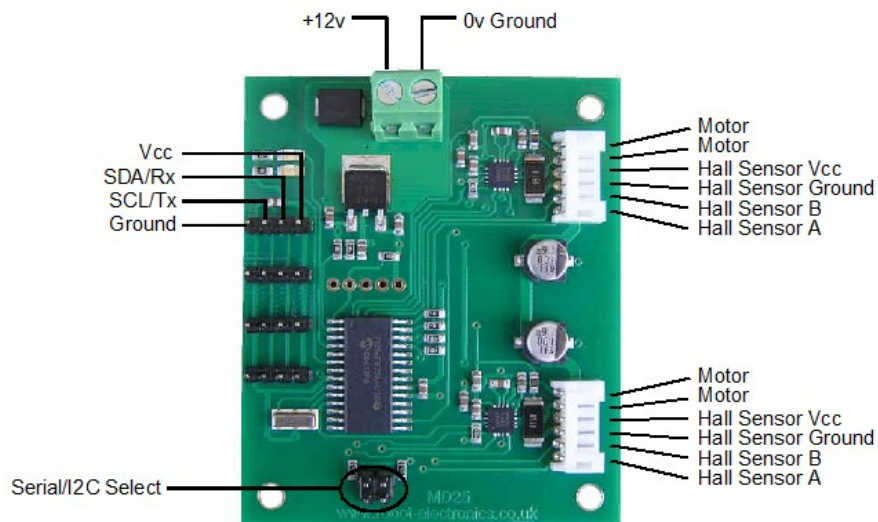


Figura V – 21. Tarjeta MD25

Teniendo la posibilidad de comunicarse mediante dos distintas formas, y tomando en cuenta que la comunicación serial está siendo utilizada por Arduino BT para comunicarse con la computadora, solo queda el protocolo I²C para comunicar. El Arduino BT cuenta con dos pines para la comunicación bajo el protocolo I²C, de forma que esta será la vía por la que se le manden comandos a la tarjeta MD25, y ésta mande los comandos de velocidad a los motores EMG30.

La idea básica consiste en acceder a cualquiera de los registros internos (véase Apéndices) de la tarjeta MD25, para transferirle los diferentes comandos y modos de trabajo, de igual forma que para leer sus variables de trabajo y su estado interno.

La manera de comunicarse consiste en enviar un registro, que identifica que se está accediendo a la tarjeta MD25. Después se envía el registro al cual se quiere acceder, dependiendo si es de lectura o escritura se envía el dato que se quiere escribir a ese registro y finalmente se cierra la comunicación con la tarjeta. Esto debe de hacerse cada vez que se quiera realizar alguna operación con la tarjeta MD25.

Para controlar la velocidad de los motores existen dos opciones diferentes a configurar, que depende de la aplicación. Para el presente trabajo se controlan los motores de manera independiente, en donde los datos de velocidad van de 0 a 255. Donde, cero indica la máxima velocidad en un sentido y 255 la máxima velocidad en el sentido contrario, por lo que la velocidad cero está determinado por el valor 128. Sin embargo, estos valores no expresan una unidad de velocidad, como la entrega el modelo cinemático. Por lo que es necesario encontrar una regla de correspondencia de la velocidad angular a la velocidad de MD25. Este problema se encuentra resuelto por un miembro del grupo de trabajo en [25], esta regla de correspondencia se muestra en la Figura V – 22.

RPM	Velocidad tarjeta	Rad/seg	Velocidad tarjeta
190	255	19.89675347	255
110	200	0	128
50	160	-19.89675347	0
-190	0		

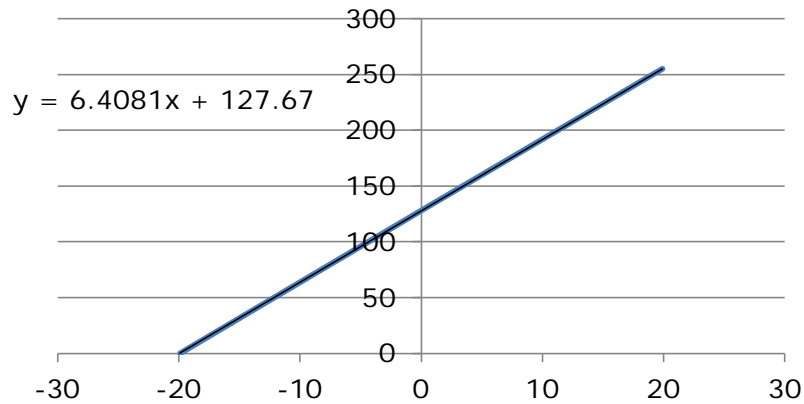


Figura V – 22. Correspondencia de velocidad angular de llantas y velocidad de MD25

Segunda parte de etapa de ejecución: Para esta segunda parte de la etapa de ejecución, fue necesario definir primeramente lo estipulado arriba, y así esta segunda parte tenga sentido. Esta parte consiste de un bloque de acondicionamiento, que se debe a lo que ya se mencionó. El modelo cinemático proporciona las velocidades de las llantas derecha e izquierda de la plataforma móvil, estas velocidades están en [rad/s] y la tarjeta MD25 tiene un rango de 0 a 255, en este bloque se programa la regla de correspondencia que se indica en la Figura V – 22. La salida del bloque son las velocidades en rango de MD25, esta señal se une a las once que se generan en la primera parte de la etapa de ejecución, volviéndose a convertir el tipo de dato a entero no signado de ocho bit. El anterior es el último bloque de acondicionamiento de datos, y finalmente los datos son enviados por puerto serial a través del bloque *Serial Send*, tal como se muestra en la Figura V – 20.

Con el párrafo de arriba se termina de describir la implementación del sistema en *Simulink*, lo único que queda es la experimentación y comprobación del funcionamiento del algoritmo coordinante de movimientos, para el manipulador móvil diseñado, lo que se presenta en el siguiente capítulo.

VI. EXPERIMENTACIÓN Y RESULTADOS.

Formando la parte final del presente trabajo, se presenta la implementación de todos los sistemas, la planeación y el algoritmo que se ha ido desarrollando a lo largo del presente. De manera que en este capítulo se describe la implementación de éste así como todos sus componentes. En una segunda parte del capítulo se describe, cómo se obtienen los datos del experimento que comprueban el funcionamiento del algoritmo y finalmente se analiza lo que se obtuvo de la prueba.

VI. 1. AMBIENTE DE EXPERIMENTACIÓN.

VI. 1. 1. ENTORNO DE PRUEBA.

La prueba se realizó en el laboratorio de Mecatrónica. En este laboratorio se encuentra un espacio dedicado a la experimentación con robots móviles, donde existe el equipo indicado para experimentar con estos sistemas.

En este espacio se ubica una cámara de video instalada en el techo del laboratorio a aproximadamente 238 [cm] del piso. Debido a la altura a la que se encuentra la cámara se genera un espacio de trabajo de aproximadamente 195 [cm], que corresponde al eje x del plano de movimiento del MM y 147 [cm] correspondiente al eje y del plano de movimiento del MM. Gracias a las medidas encontradas del espacio de trabajo, se puede escalar el sistema de visión, haciendo que corresponda a las dimensiones reales.

Por último las dimensiones de los símbolos *Fiducial*, que, de acuerdo a la altura a la que se encuentra la cámara, son rastreables para el sistema de visión, formando un cuadrado de aproximadamente 12 [cm] por lado. Donde el centro de estos, corresponde al punto detectable por *reactIVision* y que se desplaza de acuerdo a lo que se vio en el capítulo anterior. El entorno de prueba se aprecia en la Figura VI – 1.

VI. 1. 2. EQUIPO.

Sin considerar al manipulador móvil y sus elementos, el equipo utilizado para realizar el experimento consiste de una computadora, la cámara y el cable *FireWire* que comunica a estos sistemas.

La computadora corresponde a una Laptop propia del autor, modelo Toshiba Satellite M115, corriendo en Microsoft Windows 7. Procesador Intel *dual Core* a 1.75 [GHz] y una memoria RAM de 2.5 [GB]. La videocámara es de marca

Sony, modelo *Handycam* DCR-TRV 361. Existen varias formas para conectar la cámara a la computadora y que se reconozca con *reactIVision*, sin embargo la comunicación vía *FireWire* 1394 es la que permite una más rápida transferencia de datos, permitiendo una resolución de 720x480 a una velocidad promedio de 30 [fps] (cuadros por segundo). Obteniendo una imagen como la mostrada en la Figura VI – 2.



Figura VI – 1. Ambiente de experimentación.

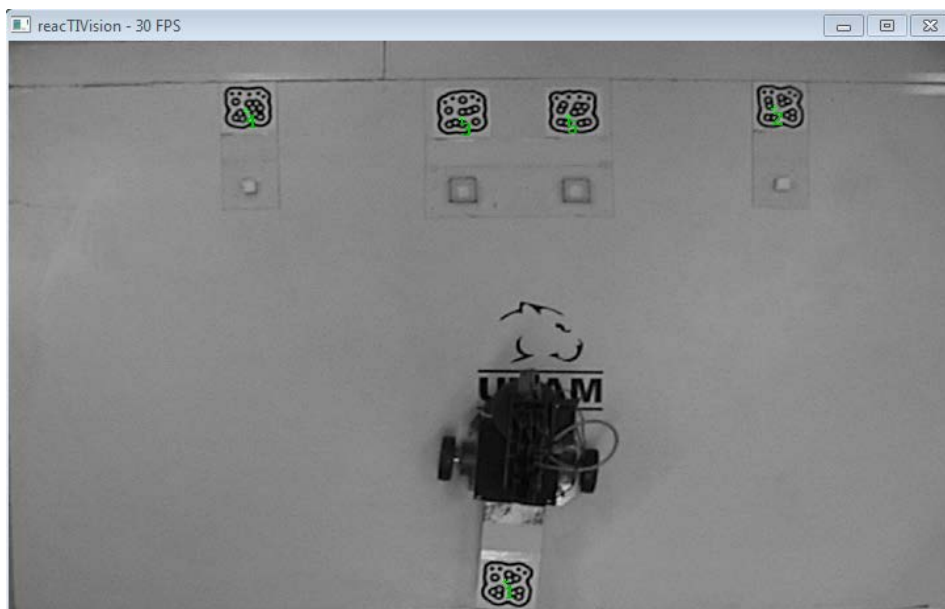


Figura VI – 2. Interfaz de reactIVision.

VI. 2. DESCRIPCIÓN DEL EXPERIMENTO.

La finalidad del experimento es comprobar la funcionalidad del algoritmo coordinante propuesto, ver su efectividad y el desempeño del robot para realizar la tarea. La descripción de la tarea, ya se ha abordado en otros capítulos, así que básicamente el experimento está diseñado a razón del algoritmo. Sin embargo, hay detalles que no se han mencionado y son necesarios para la funcionalidad del algoritmo. En la presente sección se describe de manera más detallada y sin pasar por alto algún detalle, pues todos los conceptos referentes se han descrito.

El experimento como se ha mencionado, fue realizado en *Simulink*. En la aplicación programada, se calculan todos los datos que proporcionan movimiento a ambas partes del MM y a su vez se envían por comunicación serial simulada al Arduino BT, que se encarga de mandar los comandos de velocidad de las llantas de la plataforma móvil. Todos los datos que son procesados para realizar la tarea son proporcionados por el software de visión, entrega las posiciones en el plano de objetos y robot, en base a estas se calcula todo lo mencionado. La representación gráfica se ha realizado, con la finalidad de monitorear la posición actual del MM y comprobar que todo lo enviado al robot coincida con lo que se calcula en *Simulink*. El programa de *Simulink*, la plataforma de *reactIVision* y la representación gráfica del experimento forman la interfaz de usuario. Lo que proporciona información al usuario del estado del experimento y del trayecto del MM en el espacio de trabajo. La interfaz descrita, se muestra en la Figura VI – 3.

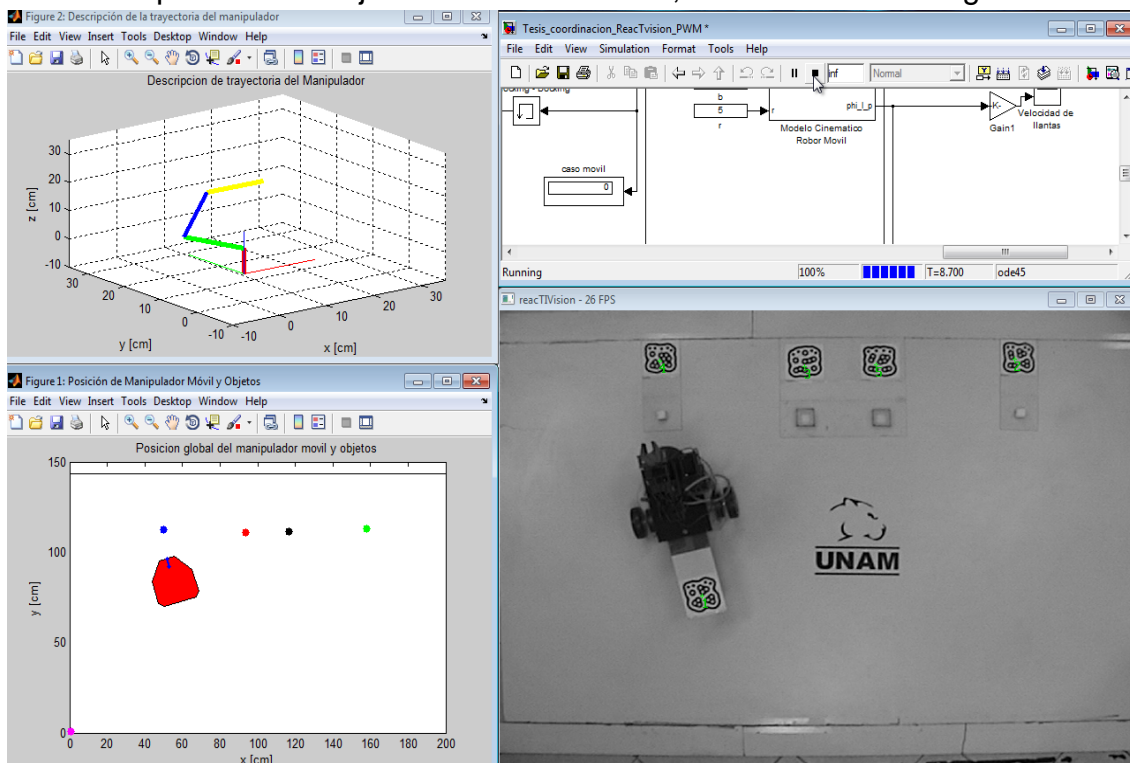


Figura VI – 3. Interfaz de usuario.

Dentro de la misma figura, se aprecia la disposición de los objetos a manipular, siendo estos los que se encuentran en contacto directo con el suelo. Por lo tanto, el brazo manipulador tiene que descender en su eje z para recoger los objetos. Al igual, los destinos de cada objeto se encuentran a nivel del piso, por lo tanto, el brazo manipulador, igualmente tiene que descender para colocar cada objeto en su destino. Se aprecia que los objetos a manipular se encuentran fijos, sin embargo, si alguno de estos se mueve la posición de *pre-docking* se actualiza, haciendo que el manipulador móvil siga el punto con tal de llegar a él, esto permite flexibilidad en el desempeño de la tarea. El identificador del MM está en el plano, que corresponde al origen del brazo manipulador, entonces significa que existe una distancia entre el identificador del MM y el piso (a), es decir, hay un desplazamiento entre la posición indicado por *reactIVision* y la posición real del manipulador móvil. Por lo tanto, la solución propuesta, con el fin de tener mayor precisión en la detección de la ubicación del MM, es una proyección del punto determinado por *reactIVision* al piso.

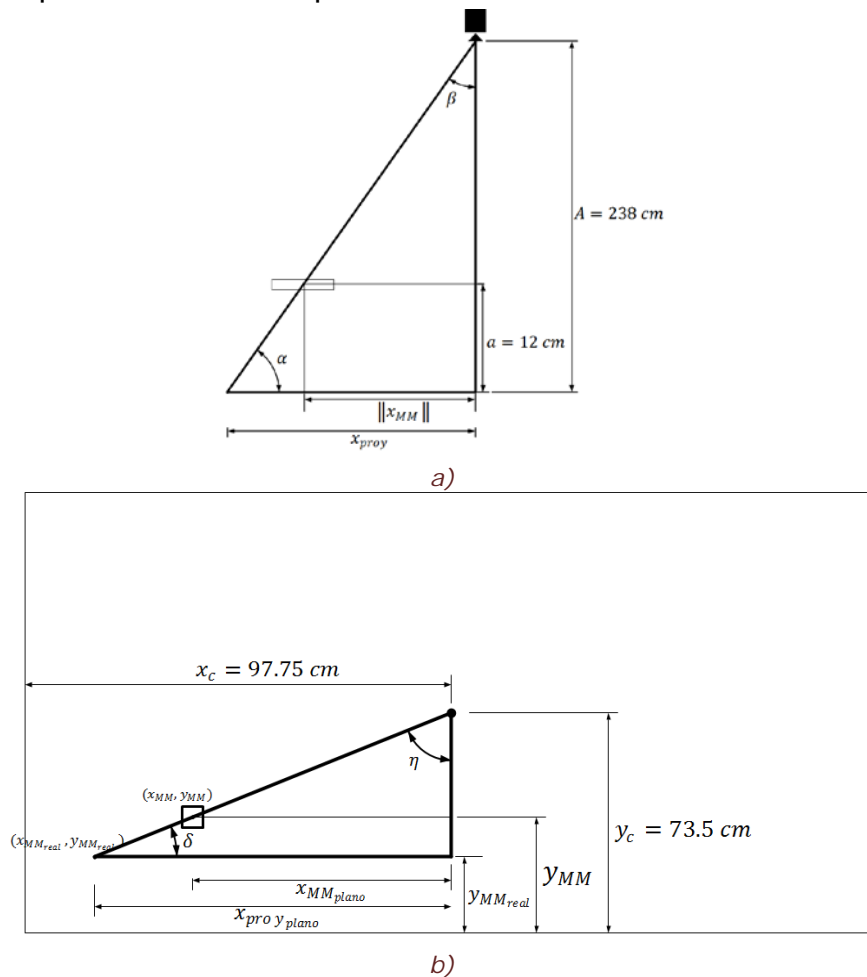


Figura VI – 4. Proyección del punto del MM al piso.

En la sección V. 5, se habla sobre un ajuste debido al entorno de trabajo, este ajuste corresponde a la proyección del punto que se muestra en la Figura VI – 4. De acuerdo a la figura, se realiza la proyección mediante el concepto de triángulos semejantes, de la Figura VI – 4a se tiene la siguiente relación:

$$\frac{x_{proy}}{\|x_{MM}\|} = \frac{A}{A-a} \quad (6.1)$$

Donde x_{proy} corresponde a la coordenada ajustada, x_{MM} es la magnitud del foco de la cámara (x_c, y_c) al identificador del MM (x_{MM}, y_{MM}) . Por otro lado para la componente en y de la Figura VI – 4b se obtiene la siguiente relación:

$$\frac{x_{proyplano}}{y_c - y_{MM_{real}}} = \frac{x_{MM_{plano}}}{y_c - y_{MM}} \quad (6.2)$$

De donde:

$$\begin{aligned} x_{proyplano} &= x_{proy} \cos \mu \\ x_{MM_{plano}} &= \|x_{MM}\| \cos \mu \end{aligned} \quad (6.3)$$

En la ecuación 6.3, el ángulo μ representa el ángulo que hay entre las coordenadas que entrega reactIVision y las coordenadas del centro del espacio de trabajo (coordenadas del foco de visión). Lo anterior se realiza para que las ecuaciones 6.1 y 6.2 tenga validez en cualquier posición del espacio de trabajo. Finalmente, se tiene que encontrar la componente en $x_{MM_{real}}$ la cual se representa por:

$$x_{MM_{real}} = x_c + x_{proyplano} \quad (6.4)$$

Con las ecuaciones 6.2 y 6.4 se obtiene las coordenadas del símbolo *Fiducial*, que corresponde al MM, proyectadas al piso. Teniendo mayor precisión en la ubicación del robot.

Para concluir esta sección se describe de forma detallada la ejecución del experimento. Primeramente, se conecta a la alimentación el manipulador móvil, haciendo que el Arduino BT se encienda. Haciendo que sea detectado por *Simulink*, entonces, el siguiente paso es abrir el programa en *Simulink*. Como el Arduino BT ya se encontraba encendido *Simulink* reconoce la existencia del puerto, permitiendo la comunicación entre ellos. De igual forma, *reactIVision* preferentemente se ejecuta antes de abrir *Simulink*, si *reactIVision* no se ha ejecutado antes de empezar con la tarea, no se envían datos, por lo tanto, el experimento no se ejecuta. Después de tener estos sistemas ejecutándose y funcionando, se ejecuta el programa en *Simulink*, que contiene todo lo que se ha descrito en el transcurso de este trabajo. *Simulink* antes de ejecutar el programa lo compila y revisa que todo esté en orden, esto hace que el inicio del experimento se tarde unos segundos en empezar.

Primeramente, mediante *reactIVision* se obtienen las posiciones y orientaciones del MM, de los objetos y de sus destinos. Los datos se ajustan, por lo que se menciona en el capítulo V y en el presente. Sabiendo dónde se

encuentran los objetos a manipular, se calcula cuál de ellos es el más cercano al MM. Una vez determinado cuál es el objeto más cercano se forma un arreglo con el orden de metas consecutivas a las que el MM debe llegar. Teniendo determinada la primera meta de cuatro, se calcula el *pre-docking*, siendo éste la primera sub-meta a la cual llegar. Debido a que el MM tiene que desplazarse en el espacio de trabajo, el brazo manipulador se posiciona en su configuración preferida y así la plataforma móvil se puede desplazar hacia la pre-meta. Cuando el punto de referencia (punto de EE) llegue a la pre-meta, se indica que hay que dirigirse a la primera meta. En esta etapa el brazo manipulador sigue en la configuración preferida. Ahora que el MM se dirige a la meta, éste se detiene cuando el radio de incidencia sea menor igual a la distancia determinada para el *docking* del MM. Cuando se indica que el robot se detuvo, lo siguiente es realizar la primera parte de la operación *P&P*.

Las coordenadas globales de visión se transforman a coordenadas del brazo manipulador, de forma que ya puede realizar la primera parte de la operación. Primero, el manipulador se desplaza en los ejes $x - y$ con la misma componente en z , que corresponde a la configuración preferida. Una vez llegando a ese punto, el manipulador se desplaza en el eje z , dirigiéndose al objeto mientras el ángulo con la horizontal incrementa, hasta encontrarse perpendicular al piso y recogiendo así el objeto. Ya que el objeto haya sido tomado, el brazo manipulador describe el mismo lugar geométrico para llegar al objeto, con la diferencia de que ahora lo realiza en forma inversa y sujetando al objeto, para no tirarlo, hasta regresar a la configuración preferida. Cuando ya se haya regresado a la configuración preferida, se indica que este evento ocurrió, haciendo que se mande la segunda meta a dirigirse, al destino del objeto que está sujetando el manipulador.

Debido a que se envía una segunda meta, se calcula un *pre-docking* de esta meta, y el MM se dirige a este punto con el brazo en la configuración preferida. Se cambia el punto a la meta, en cuanto el punto de referencia para el movimiento del robot, en el espacio de trabajo, esté dentro del área de incidencia, correspondiente al *pre-docking*. El MM se detiene de nuevo, cuando el punto de referencia esté dentro del área de incidencia, que corresponde al *docking*, en cuanto se detenga el robot, el brazo procederá con la segunda parte de la operación *P&P*. De nuevo, las coordenadas globales son transformadas a las coordenadas del brazo. La segunda parte de *P&P*, corresponde a la colocación del objeto que sujeta el brazo manipulador. Similar a la forma de recogerlo, el brazo desplaza el efector final en el plano $x - y$ a la misma componente en z de la configuración preferida. Llegando a la coordenada correspondiente del plano, al destino, el efector final descende hasta encontrarse a una distancia cerca del suelo, el destino del objeto, e inmediatamente libera el objeto, haciendo que éste

entre al compartimiento destino. Ya liberado el objeto, el brazo manipulador regresa a la configuración preferida. Como ya no hay objeto que cargar, el brazo regresa a esta configuración directamente.

Regresando a la configuración preferida, se manda la siguiente meta, la tercera que corresponde al segundo objeto a recoger. De esta meta se calcula el *pre-docking* y el MM se dirige a este. Cuando el punto de referencia indique que llego al *pre-docking*, se cambia la dirección del MM a la meta y se detiene cuando el punto de referencia indique que se llegó al *docking* del MM. Haciendo que se empiece nuevamente la operación *P&P*. De nuevo, el brazo manipulador recoge el objeto, describiendo el lugar geométrico mencionado anteriormente y se regresa a la configuración preferida.

Cuando haya regresado a la configuración preferida, se manda la nueva y cuarta meta, que corresponde al destino del segundo objeto recogido. Se calcula un nuevo *pre-docking* para esa meta y se cambia a la meta cuando el punto de referencia determine que ya llegó al *pre-docking*. El MM se detiene cuando el punto de referencia determine que está dentro del área de incidencia, correspondiente al *docking*. En cuanto la plataforma móvil se detenga, la segunda parte de la operación *P&P* se ejecuta, describiendo el lugar geométrico descrito anteriormente y libera al objeto, entrando éste al compartimiento que le corresponde. Cuando libera al objeto, se regresa a la configuración preferida y, como para este experimento no existen más objetos para recoger, el experimento llega a su fin y el MM se detiene hasta que se repita la tarea o se apague el robot.

Para este experimento, sólo son dos objetos a recoger, cabe mencionar que en el caso de que se necesite recoger una mayor cantidad de objetos, el código permite que se modifique, para cumplir con lo necesario por el usuario. Debido al limitado espacio de trabajo, sólo dos objetos a recoger son los que demuestran el funcionamiento. El fin de tener más de un objeto, es demostrar el desempeño del algoritmo y su repetitividad.

El proceso anterior describe el experimento, debido a que todo lo que se describe ya se dedujo o se calculó a lo largo del presente trabajo, la descripción está completa y el lector para este momento entiende cómo se ejecuta cada línea descrita. Sólo queda reportar los resultados obtenidos de la implementación del experimento y principalmente el desempeño del algoritmo heurístico propuesto para la coordinación de movimientos del manipulador móvil diseñado.

VI. 3. ADQUISICIÓN DE DATOS.

Se necesita reportar los datos que se obtienen del experimento, es decir, los datos que comprueben funcionalidad y efectividad del algoritmo. Por lo tanto, se tienen que adquirir los datos calculados. Debido al sistema, no se pueden adquirir datos directamente del sistema, porque la comunicación con *Simulink* sólo puede ser

unidireccional, sólo se puede recibir o enviar datos, de querer recibir y mandar datos el procesamiento del experimento se vuelve lento y no se ejecuta de forma deseada. Por consecuencia, como se ha estado mencionando a lo largo de este trabajo, sólo se envían datos a Arduino, encargado de traducir los datos enviados de *Simulink* y hacer que se mueva tanto la plataforma móvil como el brazo manipulador, según lo planeado. Entonces se tiene que los datos adquiridos, que son los que reportan la funcionalidad, están formados por aquellas señales que se obtienen de la etapa de adquisición, de procesamiento y de ejecución. En *Simulink* existe la posibilidad de adquirir datos mediante “*Scopes*” en donde entran las señales que se quieren adquirir y se guardan en el “*Workspace*” de MATLAB para su posterior procesamiento. Mediante estas características propias, de la plataforma en la que se desarrolla el experimento, se obtienen los datos a reportar.

La información más relevante para reportar es la que demuestre la efectividad y funcionalidad del experimento, pero más importante el algoritmo, los datos que se consideran son:

- La evolución de las juntas del brazo manipulador.
- La evolución de la velocidad de las llantas.
- Las posiciones en el plano x-y de la plataforma móvil y los objetos, así como el punto de referencia de la planeación de movimientos (punto final del efector final).
- La postura de la plataforma y las coordenadas cartesianas que describe el efector final.
- Los datos que definen la planeación de trayectoria y el lugar geométrico descrito por el brazo.

Siendo estos datos los que el autor considera los adecuados para demostrar su funcionamiento. En el transcurso de la experimentación, se realizaron varias pruebas, sin embargo, el experimento consiste de una sola prueba donde la tarea se ejecuta dos veces, por ser dos los objetos a recoger y colocar. El fin de realizar la tarea para dos objetos es demostrar la repetitividad y flexibilidad de la misma. Como se demuestra en gran cantidad de trabajos en el área de robótica, se pueden distinguir dos espacios diferentes: espacio de articular (de juntas) y espacio cartesiano (global), de esta forma se reportan los resultados en este trabajo, para una misma prueba, lo que permite distinguir los resultados con mayor facilidad. Los resultados reportados en este trabajo se realizan teniendo una configuración de objetos a manipular, destinos de los objetos y posición inicial del MM como la que se muestra en la Figura VI – 2. Las posiciones iniciales para la prueba reportada, que proporciona *reactIVision* son las siguientes:

$x_{MM_i} = 103.8 \text{ cm}$	$x_{o_1} = 158.1 \text{ cm}$	$x_{d_1} = 93.3 \text{ cm}$	$x_{o_2} = 49.6 \text{ cm}$	$x_{d_2} = 116.6 \text{ cm}$
$y_{MM_i} = 53.3 \text{ cm}$	$y_{o_1} = 113.1 \text{ cm}$	$y_{d_1} = 110.9 \text{ cm}$	$y_{o_2} = 112.6 \text{ cm}$	$y_{d_2} = 111.52 \text{ cm}$
$\theta_{MM_i} = 87.6^\circ$	$\theta_{o_1} = 92.9^\circ$	$\theta_{d_1} = 88.5^\circ$	$\theta_{o_2} = 86.9^\circ$	$\theta_{d_2} = 92.1^\circ$

VI. 3. 1. ESPACIO ARTICULAR.

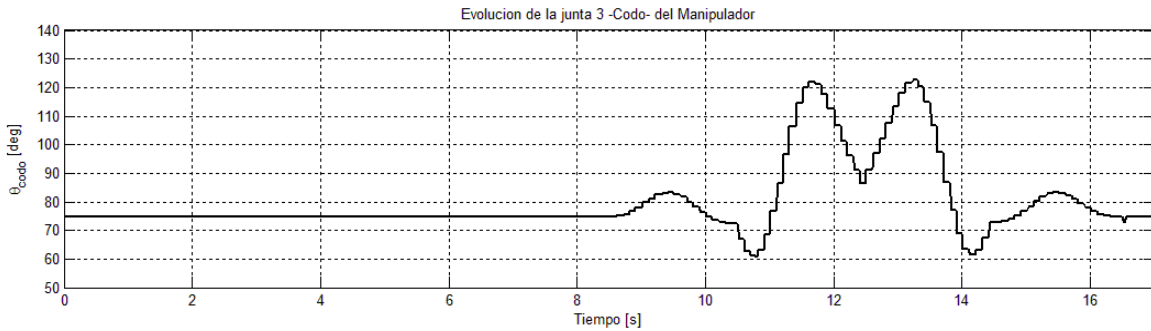
En los resultados del espacio articular, se reportan la evolución de las juntas del brazo manipulador (cinco) y el movimiento de abrir y cerrar del *gripper*. Por parte de la plataforma móvil las velocidades de cada llanta forman el espacio articular del MM. Éstas se separan en cuatro pasos que se han mencionado anteriormente: Recogiendo objeto uno, colocando objeto uno, recogiendo objeto dos y colocando objeto dos.



a)



b)



c)



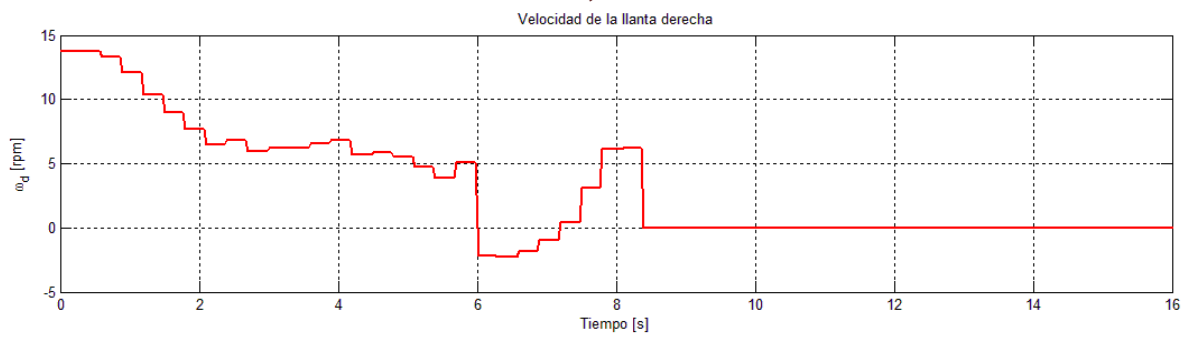
d)



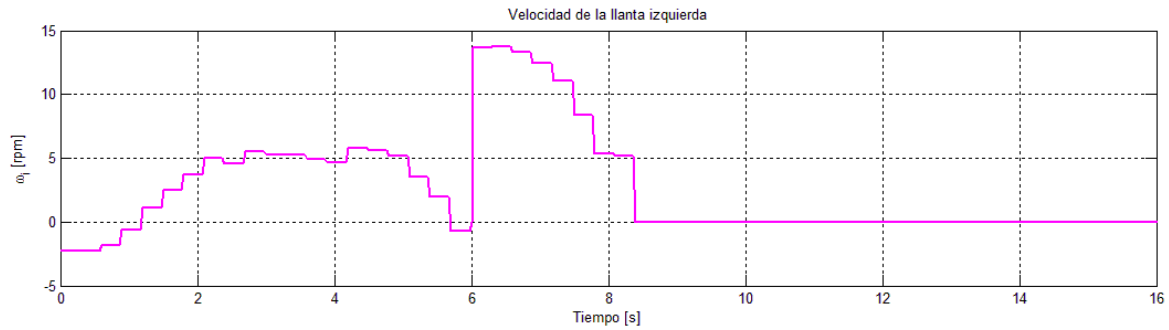
e)



f)

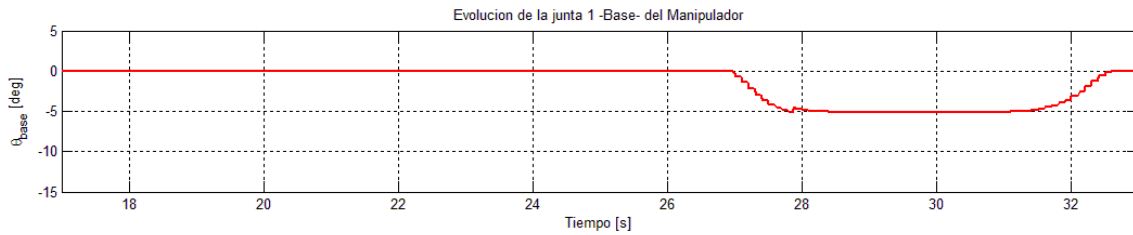


g)

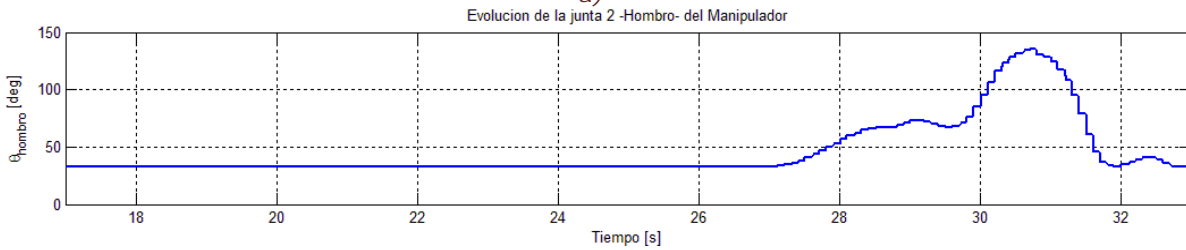


h)

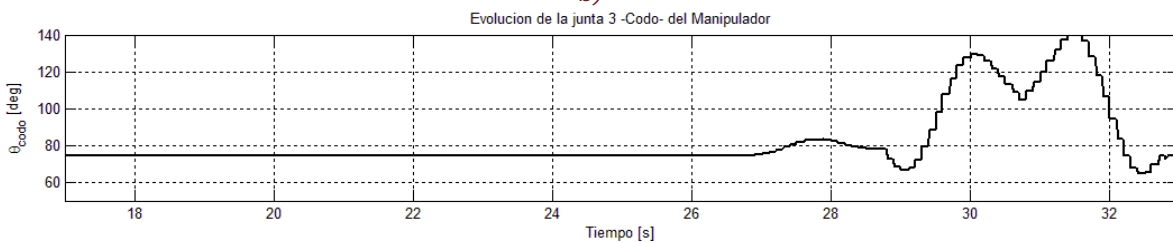
Figura VI – 5. Espacio de juntas para: Recogiendo objeto uno, a) Junta Base de manipulador, b) Junta Hombro de manipulador, c) Junta Codo de manipulador, d) Junta Inclinación de Muñeca, e) Junta Giro de Muñeca, f) Abrir y cerrar del gripper, g) Velocidad en rpm de llanta derecha de la plataforma móvil, h) Velocidad en rpm de llanta izquierda de la plataforma móvil.



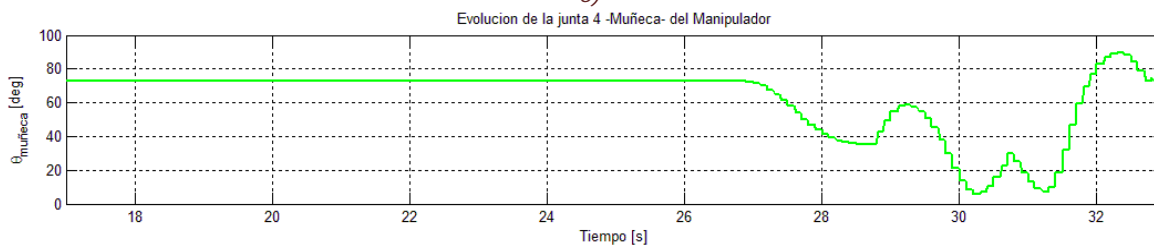
a)



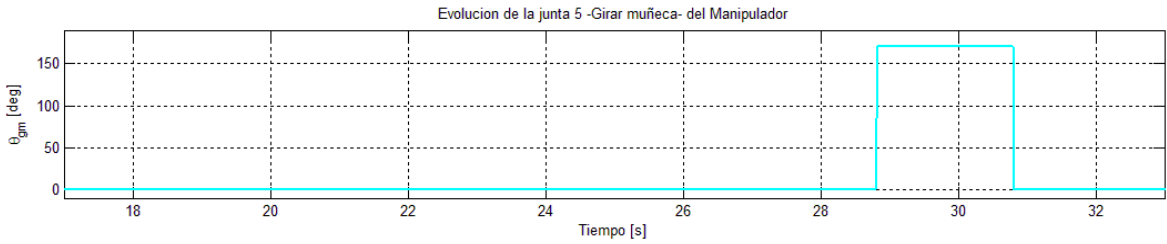
b)



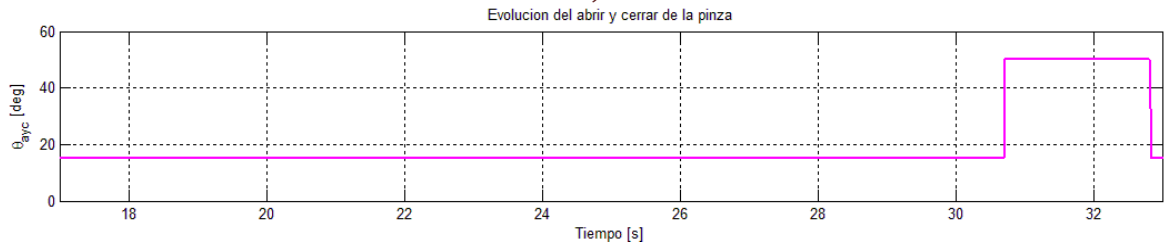
c)



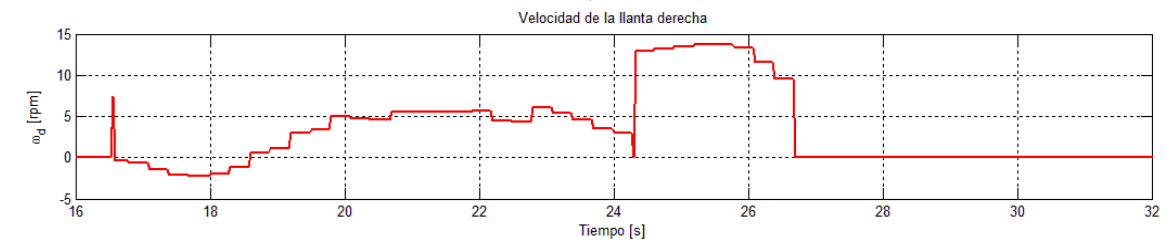
d)



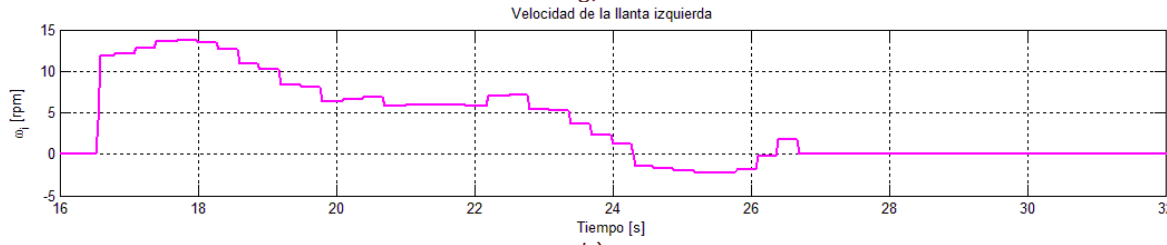
e)



f)



g)

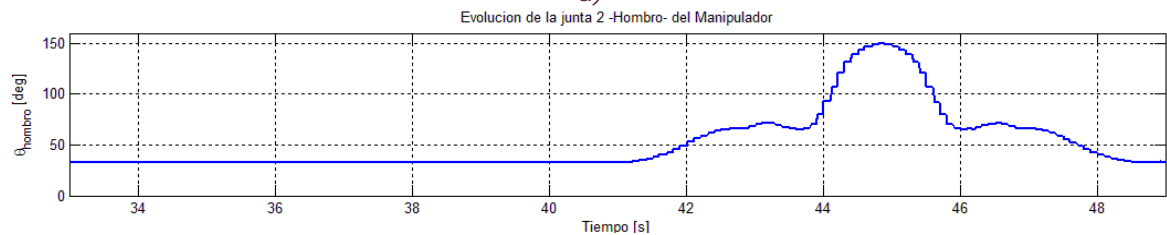


h)

Figura VI – 6. Espacio de juntas para: Colocando objeto uno, a) Junta Base de manipulador, b) Junta Hombro de manipulador, c) Junta Codo de manipulador, d) Junta Inclinación de Muñeca, e) Junta Giro de Muñeca, f) Abrir y cerrar del gripper, g) Velocidad en rpm de llanta derecha de la plataforma móvil, h) Velocidad en rpm de llanta izquierda de la plataforma móvil.



a)



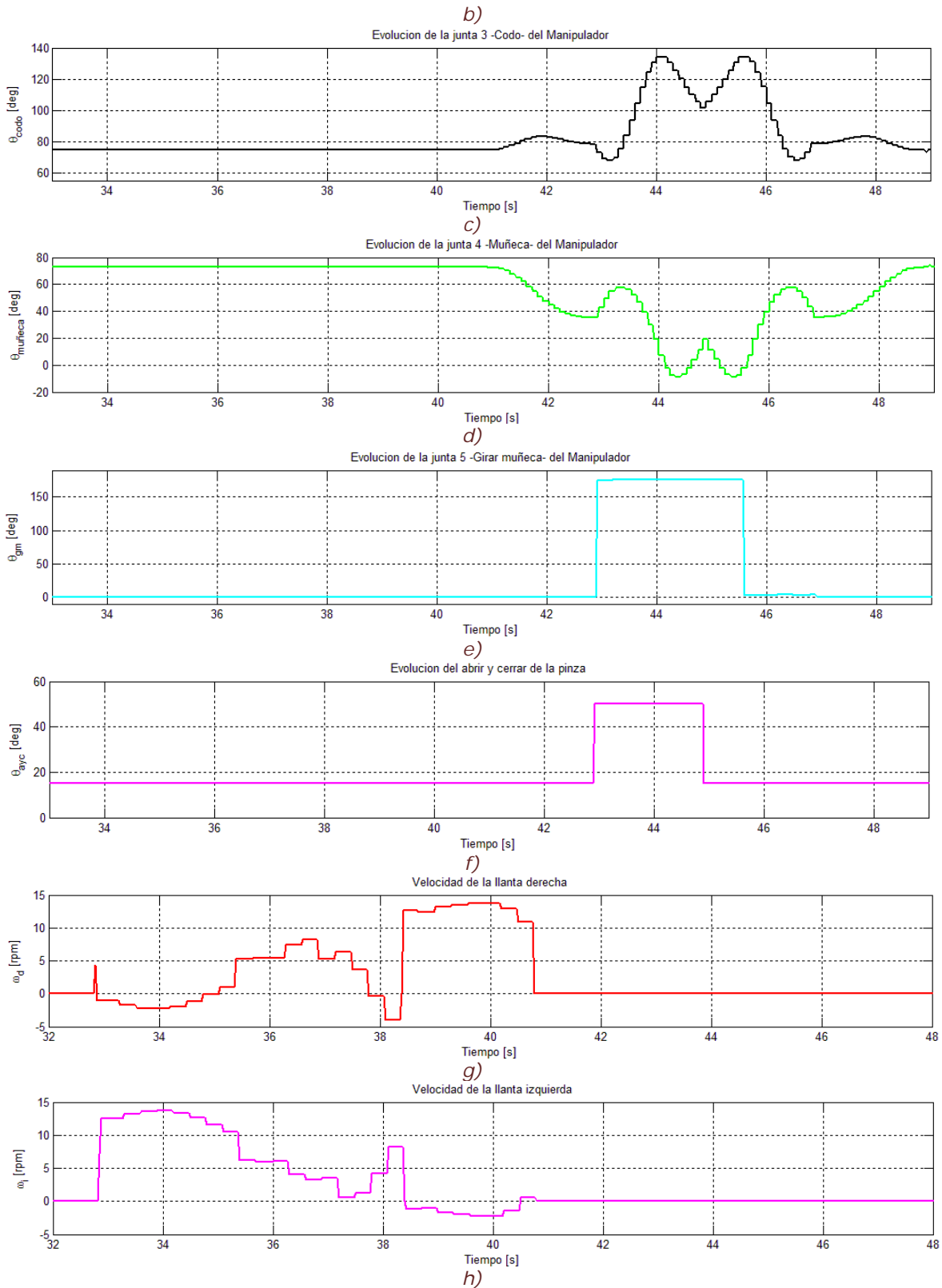
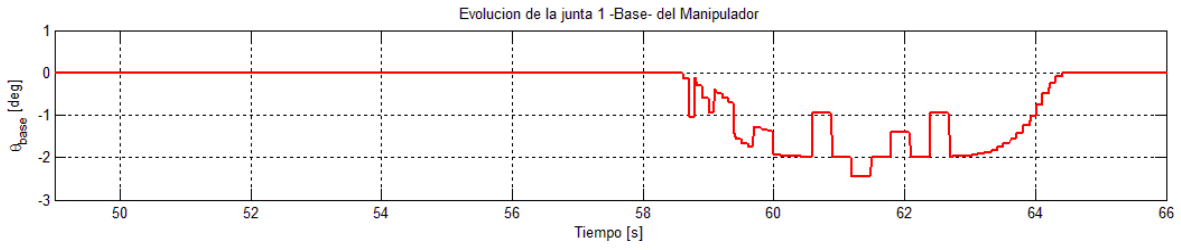
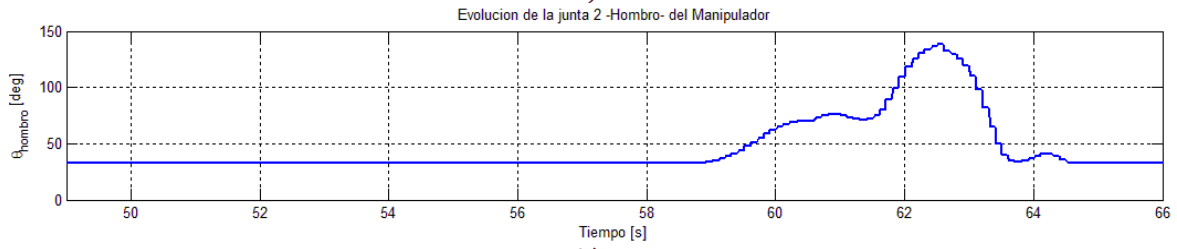


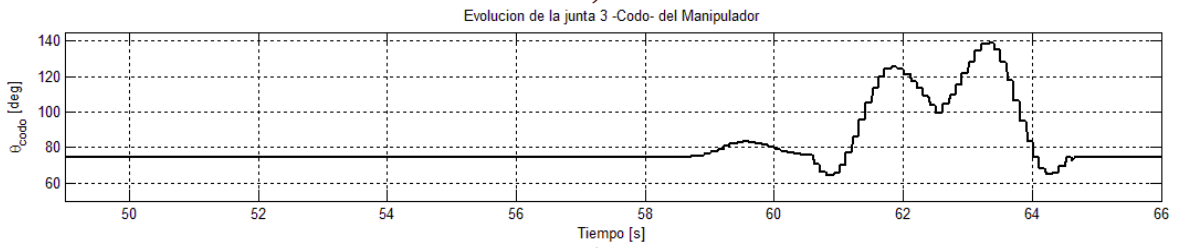
Figura VI – 7. Espacio de juntas para: Recogiendo objeto dos, a) Junta Base de manipulador, b) Junta Hombro de manipulador, c) Junta Codo de manipulador, d) Junta Inclinación de Muñeca, e) Junta Giro de Muñeca, f) Abrir y cerrar del gripper, g) Velocidad en rpm de llanta derecha de la plataforma móvil, h) Velocidad en rpm de llanta izquierda de la plataforma móvil.



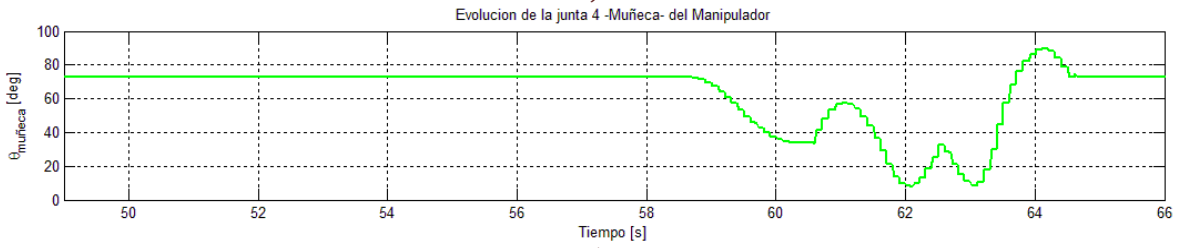
a)



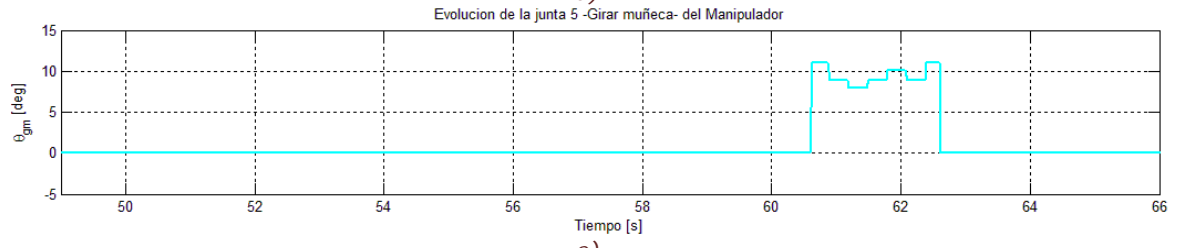
b)



c)



d)



e)



f)

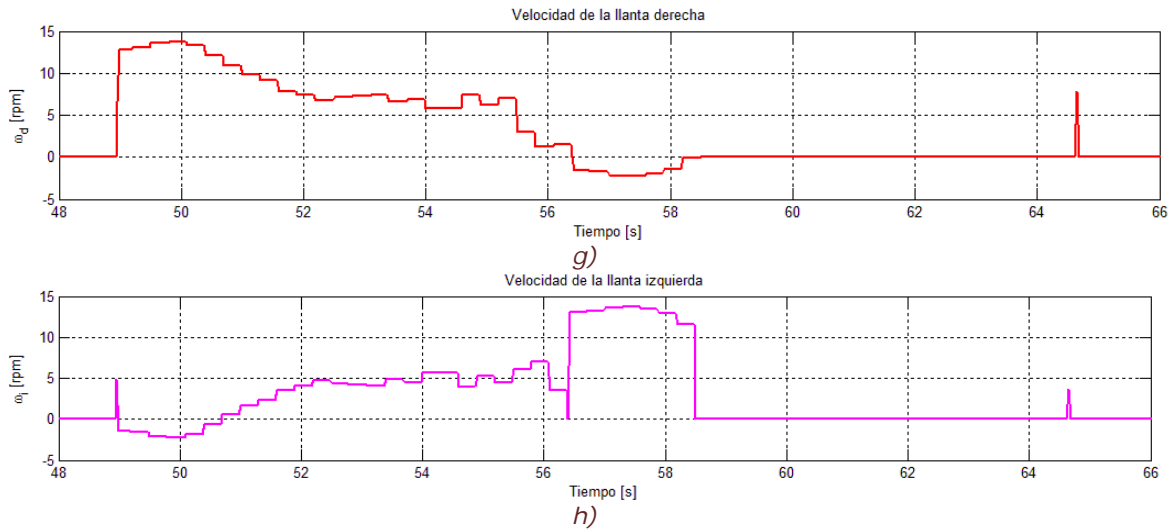


Figura VI – 8. Espacio de juntas para: Colocando objeto dos, a) Junta Base de manipulador, b) Junta Hombro de manipulador, c) Junta Codo de manipulador, d) Junta Inclinación de Muñeca, e) Junta Giro de Muñeca, f) Abrir y cerrar del gripper, g) Velocidad en rpm de llanta derecha de la plataforma móvil, h) Velocidad en rpm de llanta izquierda de la plataforma móvil.

VI. 3. 2. ESPACIO CARTESIANO.

En el espacio cartesiano, se reportan los resultados obtenidos. El espacio de trabajo del MM, está expresado en coordenadas del mundo (globales). En este apartado se muestra el recorrido de cada parte del robot, sea la descripción de la trayectoria, el lugar geométrico que describe el manipulador, básicamente todos los resultados encontrados en coordenadas globales para la misma prueba, con los mismo valores iniciales que se muestran anteriormente. En esta parte se distinguen tres diferentes pruebas: en el plano x-y, posturas de ambas partes, planeación y lugar geométrico.

VI. 3. 2. 1. PLANO X-Y.

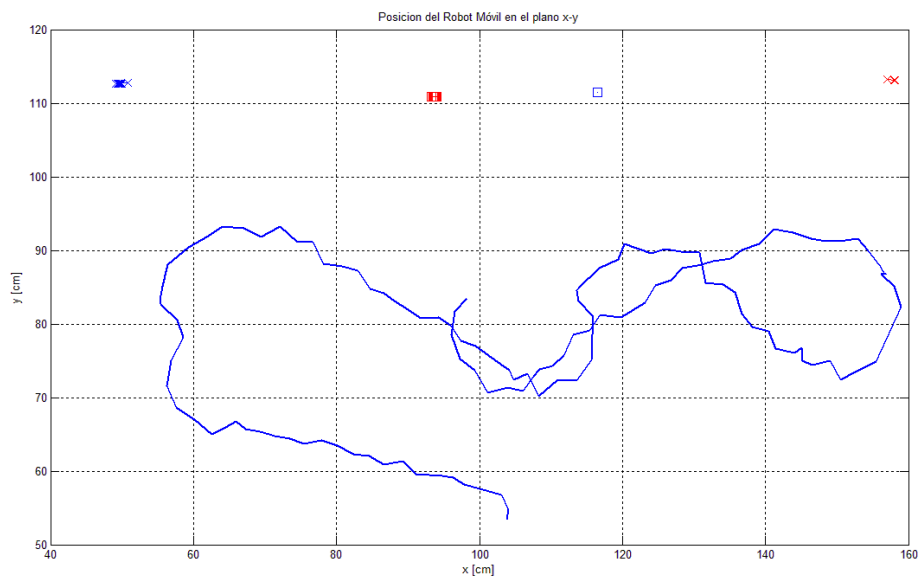


Figura VI – 9. Posición del punto P de la plataforma móvil en el plano x-y.

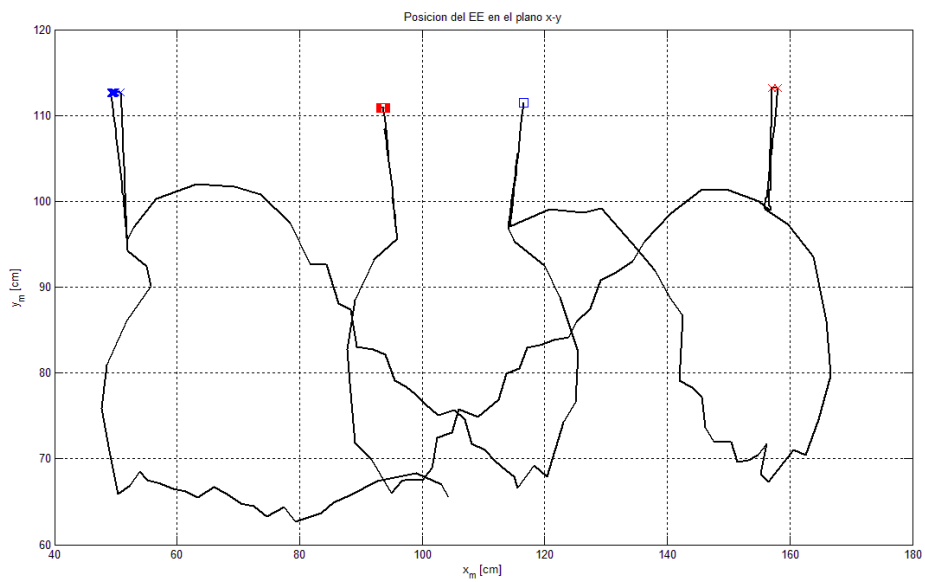


Figura VI – 10. Posición del punto p del brazo manipulador en el plano x-y

VI. 3. 2. 2. POSTURAS DE BRAZO MANIPULADOR Y PLATAFORMA MÓVIL.

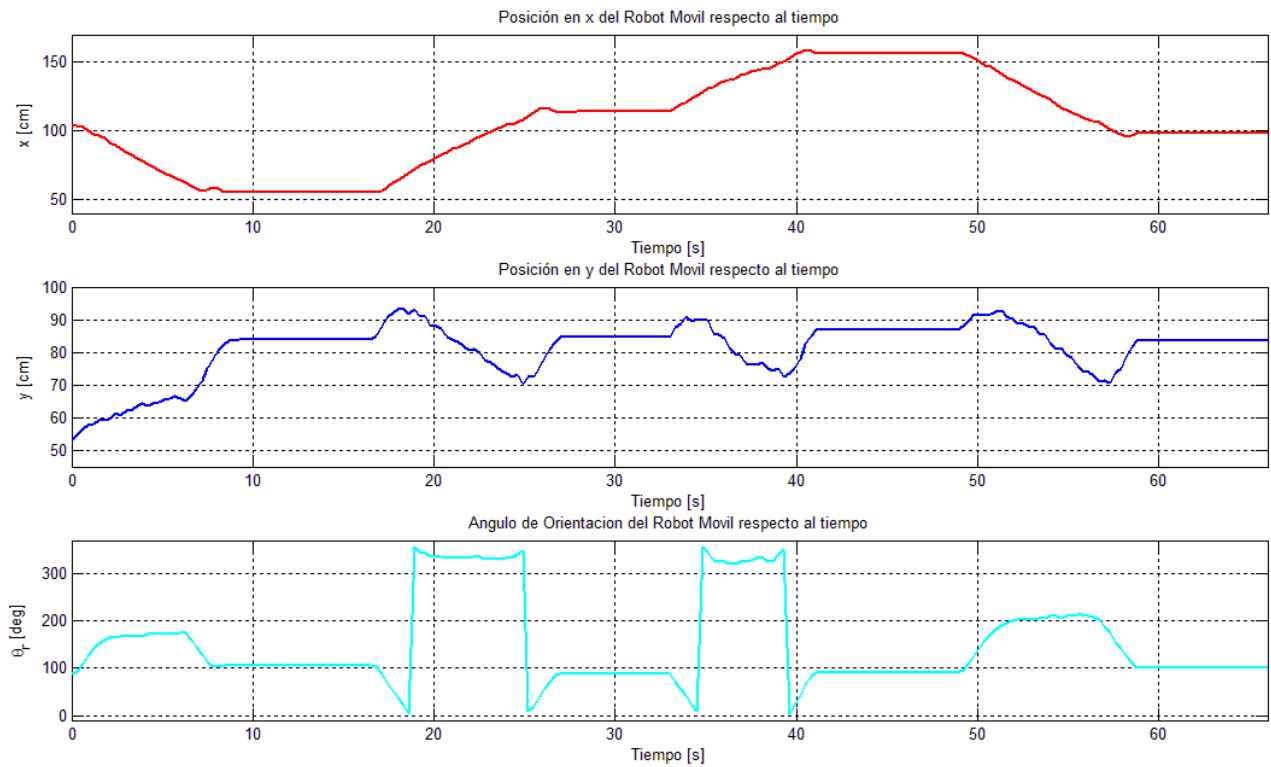


Figura VI – 11. Postura de la plataforma móvil.

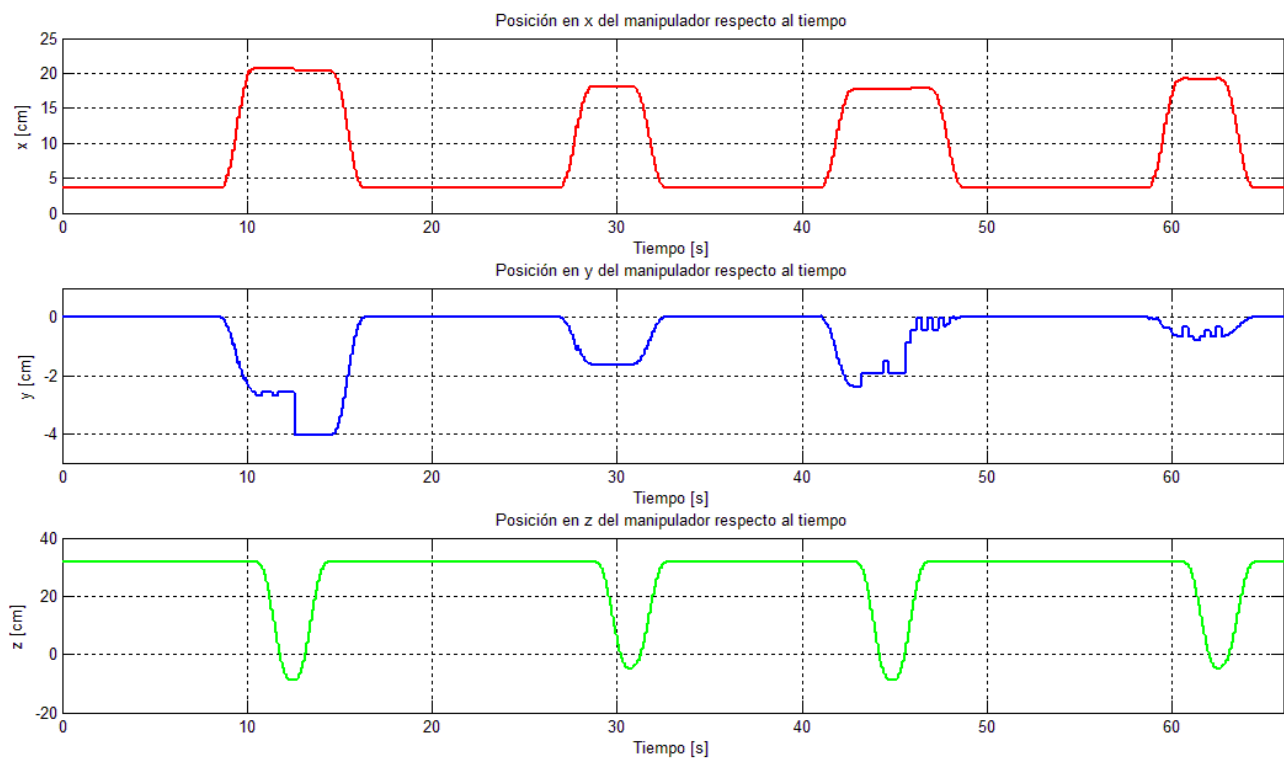
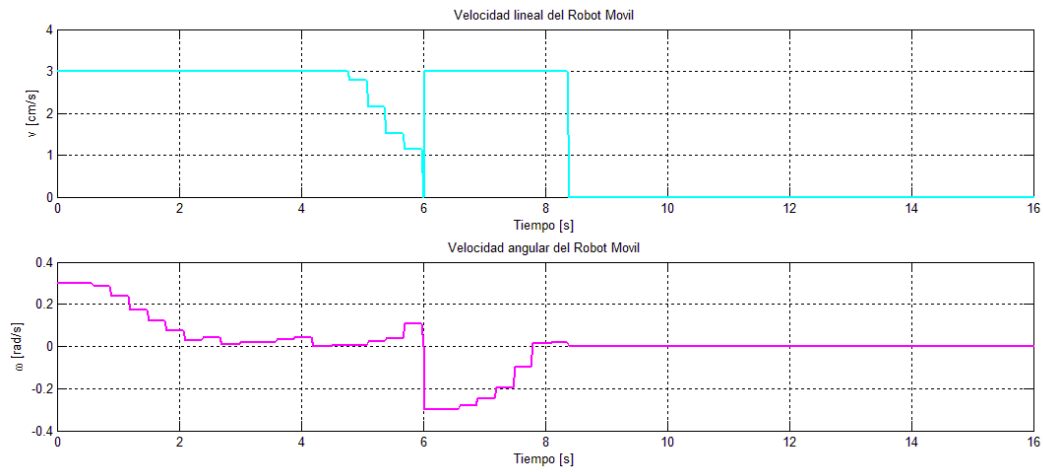
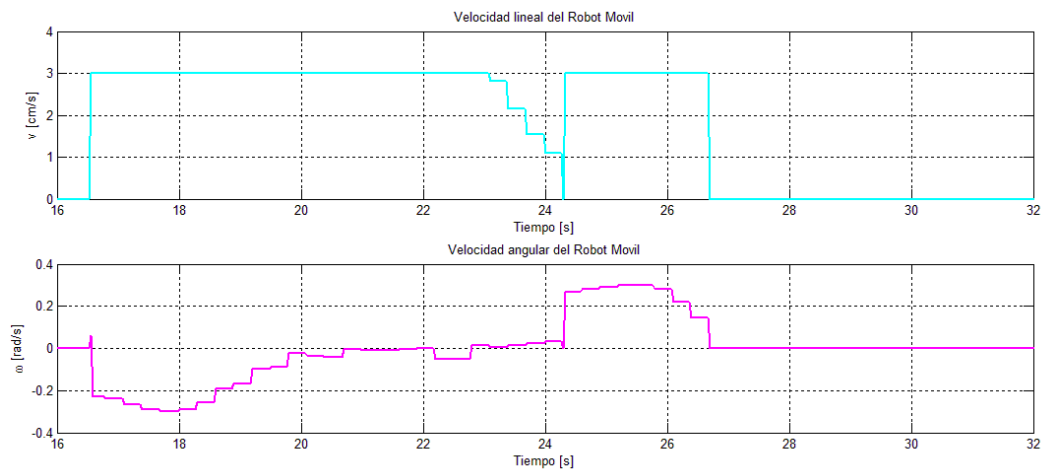


Figura VI – 12. Posición del efector final en el espacio cartesiano.

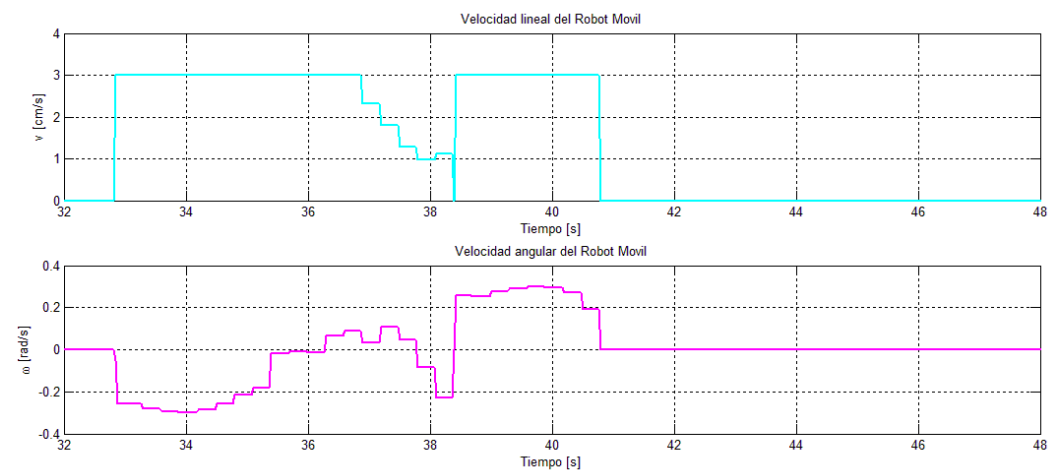
VI. 3. 2. 3. PLANEACIÓN DE TRAYECTORIA Y LUGAR GEOMÉTRICO.



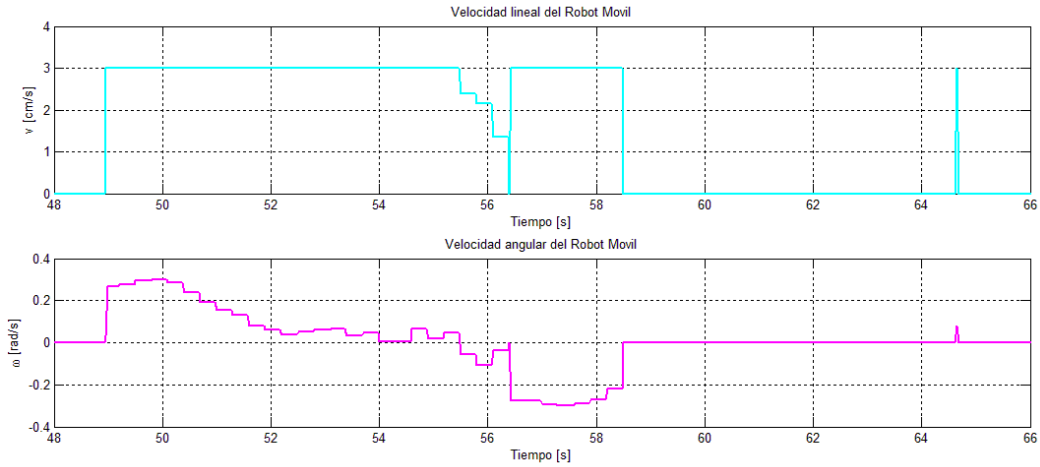
a)



b)

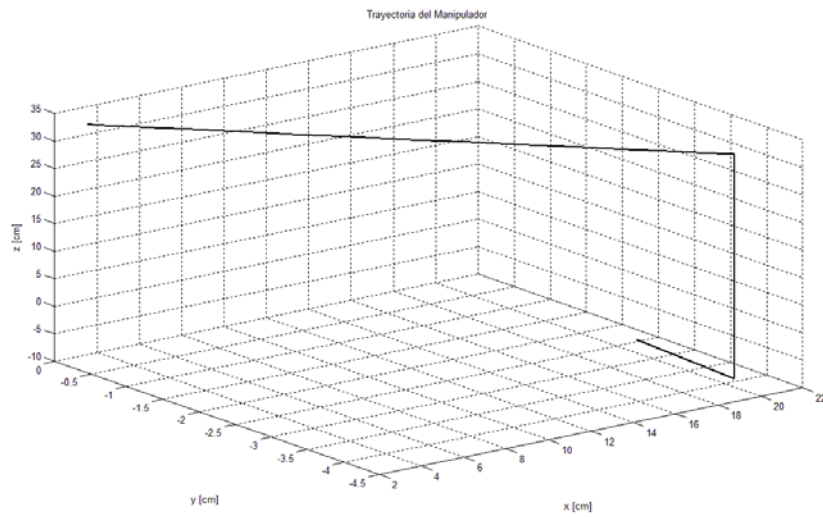
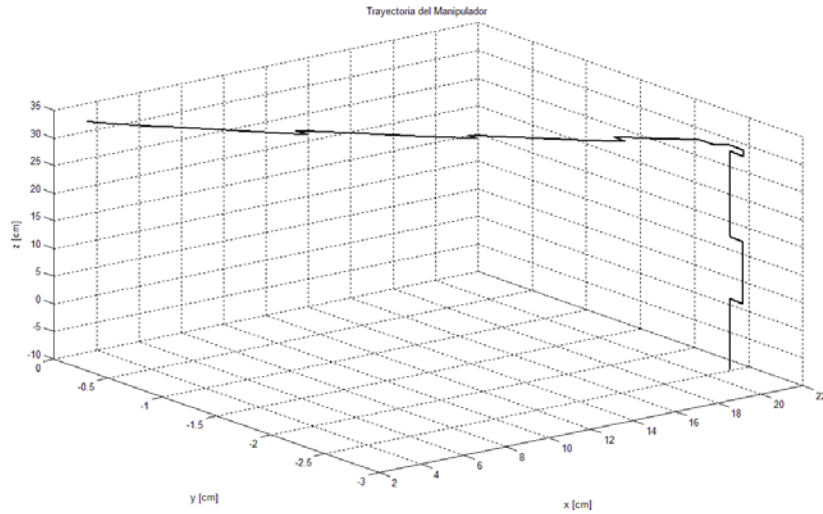


c)

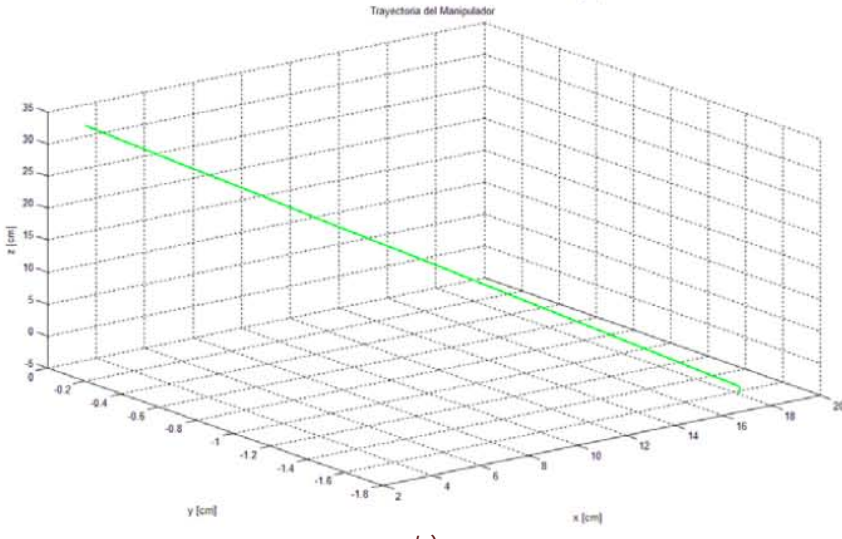
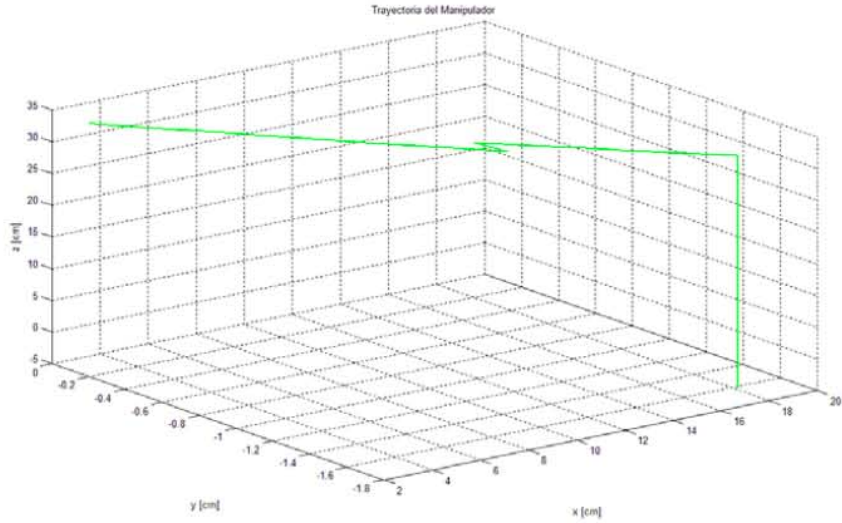


d)

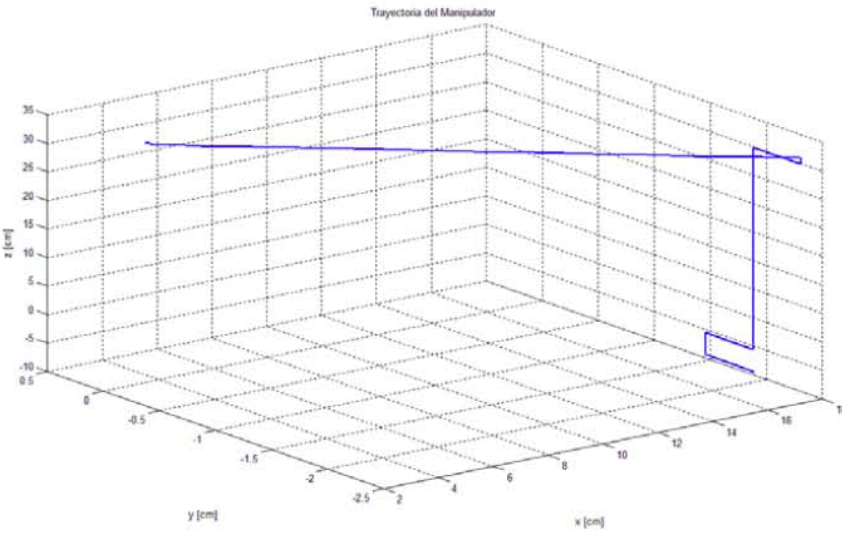
Figura VI – 13. Velocidades lineales y angulares resultado la planeaci3n de la trayectoria para la plataforma m3vil: a) Recogiendo al objeto uno, b) Colocando al objeto uno, c) Recogiendo al objeto dos, d) Colocando al objeto dos.



a)



b)



VI. 4. ANÁLISIS DE RESULTADOS.

Un hecho importante antes de avanzar, es indicar que el tiempo que se muestra es el que corresponde al tiempo de *Simulink*. Principalmente se debe, a la gran carga de procesamiento que se requiere para utilizar la visión y ejecutar todas las funciones que realizan la tarea.

VI. 4. 1. ESPACIO ARTICULAR.

En la Figura VI – 5, se muestran los valores que toman las juntas de del manipulador y la velocidad de las ruedas de la plataforma móvil al inicio del experimento, es decir, cuando el robot no trae el objeto en su poder. Se aprecia que las juntas del manipulador, se mantienen un una misma posición al inicio del experimento, lo que corresponde a la configuración preferida, los ángulos de las juntas ya se mostraron en el capítulo V. En la misma imagen los incisos a – e, corresponden a las cinco juntas del brazo manipulador y el inciso f indica cuándo se abre la pinza y cuándo se cierra. Finalmente, los incisos g y h representan la velocidad de las llantas. Se aprecia que mientras el MM se acerca a su meta (velocidad diferente a cero) las juntas se mantienen en la configuración preferida. En el momento en que el MM se detiene el brazo manipulador comienza la primera parte de la operación *P&P* para recoger el primer objeto, esto sucede aproximadamente antes del segundo nueve de *Simulink*. Antes de llegar al segundo once la junta cinco del manipulador se mueve, de forma que se orienta para que pueda recoger el objeto, esto lo realiza al terminar la primera trayectoria del brazo manipulador. También se aprecia en el inciso f que al mismo momento que la junta cinco se orienta, la pinza se abre, para recoger el objeto y se cierra cuando ya llegó a la posición del objetivo, cerca del segundo trece. Inmediatamente se vuelve a cerrar con el fin de sujetar el objeto, hasta regresar a la configuración preferida.

Al momento de concluir con la primera parte de la operación *P&P* para el primer objeto, el MM se vuelve a mover, llegando a la segunda meta, esto se ve en la Figura VI – 6 en los incisos g y h. El MM se continúa moviendo hasta llegar al *docking*, en el trayecto se aprecia, en los incisos a-d, que se mantiene en la configuración preferida y en el inciso e, la junta cinco mantiene al objeto en un ángulo de orientación cero, a la vez que la pinza se mantiene cerrada, sujetando el objeto hasta que se coloca en su destino (f). Aproximadamente, para el segundo veintisiete el MM se detiene, de acuerdo a los incisos g-h, lo que indica que el brazo manipulador debe comenzar con la segunda parte de la operación *P&P*. Las juntas del brazo comienzan a moverse hasta aproximadamente el segundo veintinueve, que termina la primera trayectoria, y la junta cinco se orienta de acuerdo al ángulo de orientación del destino, para poder colocar el objeto correctamente. Esta junta se mantiene en esa posición hasta terminar la segunda

trayectoria. En cuanto termina la segunda trayectoria, aproximadamente en el segundo treintauno, la pinza se abre, lo que permite que el objeto se libere y entre en su lugar destino. Como se aprecia en el inciso f, se mantiene la pinza abierta hasta regresar a la configuración preferida, pues conceptualmente, no trae el objeto consigo. Al momento en que se llega a la posición inicial la pinza nuevamente se cierra. Y así se termina la operación *P&P* para el primer objeto.

Debido a que este experimento consiste de dos objetos y en las Figuras VI – 5 y VI – 6 se aprecia la evolución de las juntas en el trayecto para recoger y colocar el primer objeto, en las Figuras VI – 7 y VI – 8 se muestra la evolución de juntas para el segundo objeto.

Una vez colocado el objeto uno en su destino, el MM procede a dirigirse al segundo objeto, correspondiente a la tercera meta. El MM se mantiene en movimiento como se muestra en la Figura VI – 7g-h, hacia la meta desde aproximadamente el segundo treinta y tres hasta el cuarentaiuno. Después de este segundo se indica que el MM se detuvo, lo que significa que llegó al *docking* y ahora el brazo manipulador tiene que recoger el objeto dos. Al igual para el objeto anterior las juntas empiezan a desplazarse angularmente, describiendo el lugar geométrico de la trayectoria (incisos a-d). Al terminar con la primera trayectoria nuevamente la junta cinco (inciso e) se orienta de acuerdo al ángulo de orientación del objeto dos, con el propósito de recoger el objeto correctamente y se mantiene en ese ángulo hasta regresar al punto final de la primera trayectoria. Finalmente, el inciso f indica que cuando se llega al final de la primera trayectoria, la pinza nuevamente se tiene que abrir y así se puede recoger el objeto dos. Mientras la junta cinco se mantiene orientada para recoger el objeto, se cierra al terminar la segunda trayectoria, de forma que el objeto lo tiene sujetado y así lo pueda llevar a su correspondiente destino. Después de recoger el objeto las juntas regresan a su configuración preferida y se indican que se llegó a ella.

Para finalizar el experimento en el espacio articular hay que colocar el segundo objeto en su destino, esto se muestra en la Figura VI – 8, donde una vez llegado a la configuración preferida el MM, comienza por última vez a desplazarse hacia su cuarta y última meta. El MM se continúa moviendo hasta que se detiene aproximadamente al segundo cincuenta y nueve, como se aprecia en los incisos g-h y en cuanto se detiene el brazo, comienza con la segunda parte de la operación *P&P* para el objeto dos. El brazo traza su primera trayectoria (incisos a-d) y al terminarla la junta cinco (inciso e) se orienta de acuerdo a la orientación del destino del objeto, manteniéndose en esa orientación hasta colocar el objeto en su lugar destino, que corresponde al final de la segunda trayectoria. Al final de la misma la pinza se abre (inciso f), liberando el objeto y dejándolo entrar en su lugar destino, por cualquier situación la pinza se mantiene abierta hasta regresar a la configuración preferida. Una vez llegando se vuelve a cerrar. Cuando el objeto se

libera el brazo manipulador regresa a la configuración preferida y debido a que no hay más objetos, se mantiene estático el MM y termina el experimento.

Existen varios puntos que comentar acerca de estos resultados:

- En los incisos a – d de las Figuras 5, 6, 7, 8 del presente capítulo, se aprecia que la evolución de las juntas es muy similar a simple vista, sin embargo no lo son, describen la misma forma, se debe esto al perfil de trayectoria implementado para el movimiento de las juntas. Es el mismo perfil para las dos trayectorias de ida y las dos de regreso, cuando el caso es recoger el objeto. Dos trayectorias de ida y una de regreso cuando el caso es colocar en su destino. Mirando con más detenimiento estas figuras, se ve que el rango de movimiento es diferente y es claro que sea diferente pues las ubicaciones de los objetos son diferentes. La forma en la que el MM llega a su meta es distinta, no siempre es la misma y se debe principalmente a las llantas y a la visión. Otra cosa que se puede apreciar, es que el tiempo en el que transcurre cada parte del experimento es diferente. Si se mira con detenimiento al eje correspondiente al tiempo de las anteriores figuras se verá que el tiempo de recorrido para una u otra meta no es el mismo, se aprecia que hay diferencia de varios segundos. Sin embargo, la evolución de la velocidad de llantas es muy similar en cada figura, debido a que se aplica la misma técnica para llegar a la meta en cada una de ellas, incluso las sub-metas (*pre-docking*). El mismo caso es para la evolución de las juntas del brazo, el tiempo tomado para recoger o colocar el objeto es diferente, y se debe a que el *docking* está determinado por un radio de influencia, un área. Por visión, no siempre se detendrá en la misma posición, pero lo que sí es un hecho es que la distancia entre el punto de referencia y el objeto, es decir, la distancia del *docking*, siempre será menor al radio de influencia definido.
- En el inciso f de las figuras 5, 6, 7, 8 de este capítulo, se aprecia el transcurso de la pinza (*gripper*) del brazo manipulador. Para la pinza solo hay dos valores numéricos que corresponden a cerrado y abierto (15° y 55° correspondientemente). Por lo tanto, se puede apreciar como un pulso, si está en alto está abierto y si está en bajo está cerrado, entonces cuando se encuentra en la configuración preferida, la pinza está cerrada (bajo). Se puede decir que la configuración de la pinza es estar en bajo, lo que indica que sin importar el caso, se cerrará la pinza, garantizando que no suelte el objeto, lo traiga consigo o no. Principalmente se debe a que no existen sensores que determinen si hay o no hay objeto, y así se saca provecho de no tener sensores. Además, se ahorra código, de lo contrario se introduciría otra variable para determinar, la parte de la operación *P&P* que esté ejecutando y saber si debe estar en alto o en bajo.

- En los incisos a-d de las figuras 6, 8 se aprecia que las trayectorias no son simétricas a diferencia de las 5,7 que muestran cierta simetría. Como se menciona antes y en el capítulo V, dos trayectorias de ida y dos de regreso son las que se utilizan para recoger el objeto, donde el lugar geométrico son dos rectas diferentes. Sólo que de ida y de regreso se describen la misma, es decir, la primera recta se describe y luego la segunda, en cuanto se recoge el objeto se vuelve a describir la segunda recta en sentido inverso y luego la primera en sentido inverso, de ahí la simetría. Sin embargo, cuando se coloca el objeto, se trazan dos trayectorias para colocarlo y una para regresar a la configuración preferida, por lo tanto, se describen dos rectas diferentes para colocarlo. En cuanto el objeto se libera se regresa a la configuración preferida, describiendo así una recta completamente diferente a las otras dos, con la similitud de estar formada por el final de la segunda y el inicio de la primera, lo que determina la asimetría de la trayectoria descrita.
- Finalmente, el tiempo de simulación se muestra en todas las figuras y no el tiempo real, la visión toma muchos recursos de la computadora y no permite que el tiempo de simulación corresponda al tiempo real de ejecución. En la figura VI – 8, se muestra que el tiempo total del experimento consiste de 66 segundos cuando en tiempo real el experimento consiste de 3 minutos y 13 segundos. Se podría decir que el experimento va tres veces más lento de lo que debería, sólo para aclarar la cantidad de procesamiento que se realiza en el experimento.
- En las figuras 5, 6, 7, 8 de este capítulo, se aprecia el funcionamiento del algoritmo propuesto, se muestra un buen desempeño y permite realizar la tarea. Primero la plataforma debe de llevar al brazo manipulador a la meta, mientras éste se mantiene en la configuración preferida. Cuando se llega al *docking* el brazo recoge el objeto y se regresa a su configuración preferida y lo mismo para el segundo objeto. Los resultados se describen de acuerdo al algoritmo, a cómo suceden los eventos para desempeñar la tarea. Se aprecia en las figuras del espacio articular que ambas partes del MM coordinan sus movimientos de acuerdo al algoritmo heurístico propuesto. Se aprecia que ambas partes “saben” cuándo moverse, qué es lo que deben de cumplir dentro de la tarea, es decir, que le corresponde a cada parte realizar y mediante el código se comunican ambas partes para saber si cumplen con su parte o, en el caso de que no se haya cumplido, pues se indica que aún no se ha logrado. El algoritmo de coordinación propuesto demuestra que ambas partes pueden trabajar como una sola. Aunque continúan siendo dos partes, pues cada una cumple con lo que se debe y se comunican para trabajar como un solo sistema completo.

Todo esto se aprecia a través de las juntas de cada parte: las cinco juntas del brazo manipulador y las dos llantas motrices de la plataforma móvil en el espacio articular. Solo queda mostrar los resultados en el espacio cartesiano y comentar sobre ellos.

VI. 4. 2. ESPACIO CARTESIANO.

La planeación de la trayectoria para la plataforma móvil, demuestra que se mueve como un automóvil, viendo hacia enfrente. Esto se comprueba en la Figura VI – 9, en donde se aprecia la trayectoria del punto P del MM. Del punto inicial, que se indicó anteriormente, avanza, en esta figura sólo se indica la posición en el plano, de forma que no se aprecia cuando se detiene el robot. Sin embargo, lo que se aprecia es la posición de los objetos a manipular y sus destinos, se puede ver que se mueven cuando en la prueba, estos se mantienen estáticos. No hay quién o qué los mueva, esto se puede atribuir a dos hechos: el sistema de visión no detecta la posición correctamente y se encuentra oscilando la posición en un pequeño rango de valores; la otra es que el manipulador interfiera con el sistema de visión y cause problemas al detectar las metas. Otro hecho puede ser a que los símbolos *Fiducial* no están lo suficientemente nítidos para su detección.

En la figura VI – 10, se muestra el resultado de ambas planeaciones: para la plataforma y para el brazo. El punto de operación p del EE es el punto de referencia para el movimiento del robot móvil, de ahí la trayectoria descrita por éste, cuando el brazo está en la configuración preferida, es la misma pero desplazada una distancia. La cual corresponde a la distancia entre, el origen del robot móvil y el del brazo manipulador, más la distancia del origen del manipulador a la posición inicial del brazo, aproximadamente 12 cm. En esta imagen, se aprecia cuándo el MM se detiene y se procede a recoger o colocar los objetos, según sea el caso. Partiendo de la posición inicial del punto P del MM se dirige al primer objeto, que en la figura se representa por una cruz azul, en el extremo superior izquierdo de la figura. Se nota que el robot se detiene y entonces el efector final comienza a moverse hacia adelante, indicando que se acerca a recoger el objeto uno marcado como una cruz azul. Llegando a este punto, se regresa y da vuelta a la derecha para dirigirse al *docking*. De ahí, se acerca al eje de las abscisas y de acuerdo a la imagen antes de los 120 cm en x , se dirige a colocar el objeto. Cerca de los 100 cm en y se ubica el *docking* y coloca el objeto uno en su destino. En la figura se aprecia cómo se desplaza el cuadro azul, liberado el objeto, regresa a continuar para el siguiente objeto, marcado con una cruz roja. De ahí vuelve a dar vuelta hacia la derecha y desciende de nuevo hasta los 160 cm en x llegando al pre-*docking*. Inmediatamente después, se dirige a la meta, hasta aproximadamente los 100 cm en y se encuentra el *docking* y procede a recoger el objeto dos. Cuando regresa al punto de partida gira a la izquierda y

desciende al *pre-docking* del destino del objeto, marcado con un cuadrado rojo. Aproximadamente a los 100 cm en x ubica este punto y se dirige a su meta. Cercano a los 95 cm en y se ubica el docking para el segundo objeto y procede a colocarlo en su destino. Cuando el objeto se libera, regresa al punto de partida y finalmente, el experimento ahí termina.

- Acerca de estos resultados, se aprecia que el sistema de las metas del MM a pesar de estar estáticos se mueven un poco y repercute en la trayectoria, pues para la cruz azul y la cruz roja no regresa el punto de operación p por el mismo camino que por el que llegó, por esto se ven dos líneas al momento de recoger el objeto. Lo anterior no se aprecia en los cuadrados azul y rojo debido, probablemente, a que sólo una trayectoria es de regreso. A diferencia de las dos, para recoger los objetos.
- Otra cosa apreciable en estas las figuras VI – 9-10, la trayectoria descrita por ambos puntos (P y p) no es un trazo constante, tiene muchos trazos accidentados, algunas posibles causas: las ruedas, quiere decir que hay deslizamiento, lo que no se considera en el modelo; o el sistema de visión captura posiciones erróneas de vez en cuando, sin importar la posible razón, no perjudicó el cumplimiento de la tarea, aunque sí un poco el desempeño del experimento.
- En la figura VI – 10 se muestra la coordinación del MM, pues un solo punto está describiendo la trayectoria del robot. Demostrando que las ecuaciones de acoplamiento satisfacen la coordinación de movimientos de ambas partes. Esta es una forma en la que se puede utilizar al manipulador móvil como un solo sistema.

En el espacio articular se muestran los resultados para cada junta del sistema, pero para cada sistema existe un punto que determina el movimiento de cada parte. Para la plataforma es P y para el brazo es p . Con lo que se quiere mostrar los resultados de estos puntos en específico, como lo muestran las Figuras VI -11 y VI -12. En la figura VI – 11 se muestra la postura de la plataforma móvil, aquí se aprecia el transcurso de este punto en el plano, además del ángulo de orientación de la plataforma.

- Particularmente, en la orientación se aprecian saltos desde cero hasta los 360°, esto por el sistema de visión que mide solo de 0° a 360°. Lo que se quiere decir es que si el robot gira hacia la derecha y se pasa del cero se toma que, el ángulo está en el cuarto cuadrante. Esto pueden llegar a repercutir, aunque como los resultados muestran, la tarea se ejecutó bien.

En la figura VI – 12 se muestra las coordenadas cartesianas para el EE, referidas al marco origen del manipulador. Como el manipulador (punto p) realiza un movimiento en el espacio, es pertinente mostrar la posición del punto p en las tres dimensiones. En esta imagen, se aprecia el resultado del perfil de trayectoria

utilizado lo que demuestra por qué los resultados en el espacio articular sean similares y simétricos entre sí. En esta figura se muestran los cuatro eventos ocurridos: recoger objeto uno, colocar objeto uno, recoger objeto dos y colocar objeto dos. Cuando no está manipulando, se mantiene en la configuración preferida, además aquí se muestra la posición del EE en la configuración preferida.

- En la figura VI – 12 se nota claramente cómo es afectada la trayectoria. Más importante se aprecia que la componente en y es la más afectada del problema sobre la visión. Cuando se coloca el objeto uno, no hay afectación de la trayectoria. Sin embargo, para los otros tres eventos si hay errores, particularmente al recoger el objeto uno.
- En esta figura no se aprecia mucho la coordinación, se aprecia más el algoritmo planteado. Cumpliendo con el hecho de que se mantiene en la configuración preferida y cuando se indique, le corresponde al brazo manipulador moverse para manipular el objeto.

El resultado de la planeación de trayectoria para la plataforma móvil, corresponde a la velocidad lineal y angular del punto P , por lo que al plantear esta técnica, uno puede decidir cuál es la velocidad lineal máxima y la velocidad angular máxima. El resultado se puede apreciar en la Figura VI – 13, la cual se divide en cuatro incisos, cada uno representa los eventos del experimento. El inciso a, corresponde, a recoger el objeto uno, se ve que la plataforma se desplaza a su máxima velocidad lineal, pues no se encuentra dentro del radio de influencia, en cuanto incide en este radio su velocidad decrece como se aprecia cerca del segundo seis y se detiene, indicando que llego al *pre-docking*. Luego, la velocidad incrementa de nuevo al máximo y decrece al llegar al *docking*, que inmediatamente después se detiene para recoger el objeto. En cuanto a la velocidad angular, empieza en su máximo y decrece conforme se va orientando. Al llegar al *pre-docking* gira para el sentido contrario, de ahí que parte de la velocidad angular máxima negativa y al llegar al *docking* no gira. En cuanto el objeto uno es recogido se vuelve a mover la plataforma móvil a su máxima velocidad (inciso b), hasta entrar al radio de influencia y detenerse en el *pre-docking*, aproximadamente en el segundo veinticuatro y nuevamente se dirige a la meta hasta que se detiene en el *docking*. En cuanto a la velocidad angular, muestra que gira en sentido horario (derecha) y aproximadamente en el segundo veintiuno se pasa del ángulo requerido, regresando de nuevo, girando a la izquierda (sentido anti-horario), pero como se llegó al *docking*, el ángulo de orientación es diferente y vuelve a girar en sentido anti-horario hasta llegar al *docking* y dejar de girar. Cuando el objeto se coloca la plataforma móvil, se desplaza una vez más dirigiéndose al *pre-docking* del segundo objeto (inciso c), repitiendo el mismo proceso: mantenerse en la velocidad lineal máxima y decrece

ésta en cuanto se incida en el radio de influencia, llegando al *pre-docking* y dirigiéndose a la meta. Cuando incida en el radio de influencia que determina el *docking* se detendrá. Mientras que la velocidad angular, orienta a la plataforma lo más posible a su meta o sub-meta, posible debido a que el *pre-docking* y *docking* determinan qué tanto llegue a orientarse, estos puntos indican, cuándo hay que detenerse. Finalmente, se dirige a la última meta (inciso d), realizando el mismo procedimiento descrito.

- Acerca de los resultados hay que comentar que la velocidad lineal máxima y la velocidad angular máxima son propuestas por el autor y que determina el mejor resultado. Si las velocidades son mayores a las propuestas el sistema de visión pierde el símbolo *Fiducial*, haciendo que el experimento falle, lo cual no se debe permitir. Las velocidades indicadas se muestran a continuación:

$$v_{max} = 3 \frac{cm}{s}$$

$$\omega_{max} = 0.3 \frac{rad}{s}$$

- La dirección a la cual gira la plataforma, la determina el ángulo de error establecido en el capítulo V, para la planeación de trayectoria. Por ser función de la función trigonométrica seno, que adquiere valores [-1,1], y el signo de este indica hacia donde tiene que girar.
- El hecho de pasarse del ángulo correspondiente se atribuye probablemente a dos hechos: la velocidad del robot continua siendo mayor a la que debería, por los parámetros de diseño o debido al sistema de visión; pues cuando detecta la primera posición, no ha llegado al ángulo, pero al volver a detectar este punto se pasa del ángulo y tiene que girar en sentido contrario para compensar.
- En esta figura se demuestra el funcionamiento del algoritmo, se aprecian los cuatro eventos destacados, lo que representa la funcionalidad del mismo. Se separan bien los momentos en que cada parte desempeña su labor y se aprecia que trabajan en su momento indicado.

Para finalizar los resultados obtenidos se reporta el lugar geométrico descrito para la operación *P&P* en la Figura VI – 14. Al igual que antes se distinguen los mismos cuatro eventos, solo que ahora para el punto de operación *p* del EE. Primero, en el inciso a, se muestra que el EE parte de la configuración preferida, desplazándose hasta colocarse en el mismo punto en *x – y* y a una *z* inicial. Llegando a este punto se traza la siguiente trayectoria, que es descender para recoger el objeto uno, el punto *p* del EE llega hasta una cota de $z_{frecoger} = -9 \text{ cm}$. Llegado a este punto, se regresa al punto inicial describiendo las mismas dos trayectorias en sentido inverso. En el inciso b, se muestra el lugar geométrico

descrito para colocar el objeto en su destino. Primero se trazan las dos trayectorias arriba mencionadas, con la diferencia que para colocar el objeto la cota del punto p del EE desciende hasta $z_{f_{colocar}} = -5 \text{ cm}$. Esto quiere decir que el objeto se libera desde una posición alta, relativamente, así permitiendo entrar al objeto en un depósito que coincide con la geometría del mismo. Cuando el objeto sea liberado, se regresa a su posición inicial y como muestra la figura, en el inciso b, la trayectoria se traza desde la posición final hasta el punto correspondiente a la configuración preferida. En el inciso c se muestra el mismo proceso mencionado arriba, sólo que para recoger el objeto dos. Se aprecia que la trayectoria es la misma, sin embargo, el lugar geométrico es diferente pues son diferentes ubicaciones del objeto, compartiendo la misma posición de partida y la cota final correspondiente para recoger un objeto, al igual se muestra la trayectoria de ida y la trayectoria de regreso. Finalmente, en el inciso d se muestra las trayectorias para colocar el objeto, igual que para el objeto uno, consiste de la trayectoria de colocar, formada por dos rectas, donde la cota final de la segunda recta corresponde a la misma que se menciona arriba para colocar un objeto. La trayectoria de regreso, dirigiéndose de la posición final al punto inicial, correspondiente a la configuración preferida. El hecho de realizar más directo el regreso es porque no se está cargando el objeto y no hay nada que se pueda soltar y perjudique la funcionalidad del experimento, además de ahorrar algunas operaciones y variables.

- Lo que se aprecia de esta figura, es que el lugar geométrico descrito no son realmente líneas rectas. Debido a las diferentes posiciones que manda *reactIVision*, ya se ha mencionado este problema que se presenta de una u otra forma en los resultados presentes. Una situación así no se consideró al momento de realizar el experimento, en el código no se encuentra algún filtro para esta situación, lo que pudo en dado momento contrarrestar estos efectos. Esto es lo que se aprecia en la figura, principalmente en el inciso d, donde el lugar geométrico no son rectas sino segmentos de ellas, que debido al cambio de la posición se calculan para las nuevas posiciones formando varios segmentos de recta en un mismo plano, reiterando que una situación así no se consideraba. Esta situación sí afecta la funcionalidad del experimento, sin embargo, esta figura representa únicamente la trayectoria del EE del brazo manipulador, no forma parte del algoritmo de coordinación ni de la coordinación de movimientos, esta parte le corresponde al brazo únicamente realizarla. Si ambas partes se movieran al mismo tiempo esto repercutiría totalmente en la coordinación de movimientos.

VII. CONCLUSIONES Y TRABAJO FUTURO.

Discusión:

En el presente trabajo, se han mostrado las diferentes técnicas utilizadas (campos potenciales, operación *P&P*, el algoritmo heurístico de coordinación, entre otras), y todas las soluciones propuestas para lograr que el Manipulador Móvil diseñado que se compone de plataforma móvil y un brazo manipulador serial, logre que ambas partes trabajen de manera coordinada, con el propósito de cumplir la tarea diseñada.

La tarea diseñada es sencilla, pues la finalidad del trabajo es mostrar el algoritmo de coordinación y permitir que el manipulador móvil realice la tarea coordinada. Con el algoritmo implementado en este trabajo, se pueden realizar tareas más complejas que pueden demostrar otras teorías. Sin embargo, cabe la posibilidad de cambiar la técnica de coordinación (configuración preferida), gracias al acoplamiento implementado; lo que quiere decir que, la técnica puede ser cambiada. El experimento diseñado en *Simulink* permite que el código que representa está técnica se cambie. La técnica se representa en el bloque de "Movimiento Manipulador", que cambia teniendo en cuenta las entradas y salidas correspondientes para el resto del experimento, cuidando mucho los indicadores que forman parte del algoritmo. También se puede crear un nuevo experimento para el robot diseñado.

En el capítulo III, acerca del diseño del robot, se menciona que se trata de la primera iteración de una etapa de diseño más amplia. El Manipulador Móvil utilizado para realizar el experimento es un prototipo funcional. Sin embargo, existen cuestiones sobre el diseño que deben considerarse en el futuro que refieren a que, el diseño debieron hacerlo dos o más personas. Con el propósito de que las ideas de un miembro se refuten por el otro, o que las decisiones sobre los elementos a emplear se tomasen en conjunto. Así, las cuestiones que le pasaron por alto, al único autor del presente diseño, sean salvadas con dos o más personas que acuerden. Este diseño presenta la característica de ser flexible pues es modular, cualquier sistema puede ser remplazado por otro, formando un nuevo sistema híbrido, ó, utilizando cada sistema original. El más moldeable es el efector final, pues en este trabajo es una pinza que sujeta los objetos a manipular, pero en caso de ser necesario, puede cambiarse por otra herramienta, para desarmar piezas o realizar cualquiera de las múltiples tareas de manufactura. Existe también el factor económico, el robot funciona pero, la junta del codo se encuentra en el límite de carga, por esto es que los objetos a manipular están hechos de unicel,

cuando originalmente eran de nylamid. Al hacer las primeras pruebas, se encontró que el brazo no soporta la carga representada por el nylamid, así que se realizaron los objetos con unicel. El servomotor que otorga movimiento a la junta del codo se tiene que cambiar por uno de mayor par, que supere el par calculado inicialmente. Y he aquí que se encontró una limitante económica, pues los servomotores de alto par cuestan, del doble hasta el cuádruple del utilizado en este robot. Se demuestra que funciona, pues a pesar de que la junta del hombro está formada por dos servomotores iguales al del codo, se pudo conseguir un servomotor de alto torque con un valor poco menor a cuatro veces el original y así llevar a cabo las modificaciones pertinentes. Otra posible solución, es que se realice el robot más pequeño y evitar el problema de la limitante de carga.

La forma propuesta para coordinar los movimientos muestra buenos resultados, ya que, sin importar donde se encuentre el robot, primero debe orientarse para manipular el objeto asignado, y esto, compensa la limitante de movimiento del brazo manipulador. Esto demostró que las dos partes del Manipulador Móvil trabajaron en conjunto. El brazo manipulador cuenta con un espacio de trabajo al estar fijo y la plataforma móvil, lleva al sistema hasta el *docking*, que representa una distancia dentro del espacio de trabajo del brazo, hecho que comprueba trabajo coordinado. La operación *P&P*, es una tarea muy básica para los sistemas manipuladores, y se tiene la libertad de definir el lugar geométrico que desee, esto permite que la operación se haga tan fácil o tan compleja como se desee. Con la finalidad de demostrar el funcionamiento del algoritmo, este se planteó lo más sencillo posible, con líneas rectas para llegaran al objetivo, por ende, cumplir con la operación. La principal razón de realizar la trayectoria con un perfil, es evitar que el brazo alcance una velocidad alta y que su inercia, no desarme, ni destruya el brazo, pues es de acrílico y no resiste grandes golpes. Sin embargo, el perfil de trayectoria en sus características originales, se realiza con tiempo de ejecución. En este trabajo, se aplicó la esta técnica con un contador de puntos, que va formando la trayectoria con cierto número de discretizaciones. Que la trayectoria se realice de esta forma hace que el movimiento del manipulador sea pausado e inconstante. El principal factor de este hecho es que al momento de realizar el experimento no se encontró una solución en *Simulink*, para que iniciare el tiempo en cuanto el brazo tuviera que manipular el objeto, tal y como la tarea lo describe: el manipulador se mueve hasta llegar al *docking*. Es decir, la trayectoria por medio de puntos entrega buenos resultados, a pesar de que dé la impresión de realizar el movimiento entrecortado. El desplazamiento del MM en su entorno de trabajo, se realiza considerando un ambiente ideal, sin perturbaciones, sin objetos que interrumpen su camino. Esto fue considerado para que el experimento no se complicara y no se sobrecargara el procesamiento de la computadora. Es preciso que el entorno de trabajo del MM, es demasiado pequeño como para agregar más sistemas, debido principalmente

al gran tamaño del robot y, a que hay que hacer arreglos al sistema para que la visión entregue resultados satisfactorios.

Acercas del sistema de visión, se aprecia que este tipo de vision no es la mejor solución para un sistema como el que se reporta en este trabajo, pues el tamaño del manipulador en movimiento, interfiere en el campo de visión de la cámara, provocando que los símbolos *Fiducial* se pierdan y el experimento fallé. Los inconvenientes encontrados en este experimento relativos a la visión fueron los siguientes:

- Los símbolos *Fiducial* parecieron pequeños. Se propone que deban ser muy grandes para que el sistema de visión los detecte sin problemas, de esta manera se resta espacio al entorno de trabajo. Como se muestra en el capítulo V, la ubicación del punto de trabajo se tiene que desplazar cierta distancia para detectar el objeto.
- El hecho de que el desplazamiento del punto de trabajo reduzca mucho espacio (de 18 cm para los objetos y casi 30 cm para el MM), impide agregar más sistemas al experimento.
- Y otro factor importante es el entorno de trabajo, que está definido por la altura de más de 200 cm, a la que la cámara se encuentra, formando un espacio de trabajo amplio para robots pequeños pero, no para el robot utilizado en este trabajo cuya principal limitante es el techo del recinto.

Conclusiones:

Por lo tanto se pudo concluir que el sistema de visión empleado en este trabajo, no es el mejor para este tipo de robots híbridos, las mejores soluciones se pueden apreciar en el capítulo II, donde se muestran MM existentes y que cuentan con el sistema de visión montado al mismo MM. Modificando el algoritmo propuesto y con la adición de más sistemas al robot, podría obtenerse otra solución.

El diseño del experimento en *Simulink* otorga buenos resultados, demuestra que el algoritmo se puede implementar y que el Manipulador Móvil de este trabajo coordina los movimientos de ambas partes satisfactoriamente. No obstante, se puede mejorar el experimento. El hecho de no poder adquirir datos del robot, porque en *Simulink* la comunicación bidireccional retarda el procesamiento del experimento, obtener datos del exterior, permitiría que el experimento tuviera la posibilidad de compensar errores de la prueba real, mejorando el desempeño. Una posible solución sería mudar a plataformas como *C#*, que se tendrían que modificar y adaptar a esta plataforma.

Existen algunas observaciones e ideas, a posteriori, que pueden modificar el experimento propuesto en este trabajo, se pueden desarrollar soluciones para continuar explorando los temas abiertos en esta tesis, lo cierto es que este robot, el algoritmo y todo lo que aquí se ha propuesto, puede formar parte del trabajo futuro en este ámbito. En los siguientes párrafos se sugieren algunos temas a desarrollar en el futuro.

Trabajo a futuro:

Se sugiere, con base en los datos obtenidos en este experimento, que la dimensión de piezas, ranuras y orificios sean más adecuados y se considere que los elementos de unión sean estandarizados. La pieza más importante a modificar, es este trabajo, fue el *gripper*, pues debe tenerse en cuenta que, tenga la posibilidad de cerrar completamente los dedos y recoger objetos más pequeños.

En este modelo hay que cambiar los elementos que proporcionan movimientos, principalmente para el brazo. Entre las futuras cosas por hacer son:

- Cambiar las piezas que forman la junta del hombro para que se utilice un solo un solo servomotor. Se debe conseguir un servomotor de un alto par para evitar en lo posible, la limitación de esta junta, sin generar gastos innecesarios.
- Cambiar el servomotor que forma la junta del codo, por uno de alto par, que supere el límite para la carga establecida en el capítulo III. Así, ya no habría problemas al manipular un objeto de mayor peso.
- Modificar la junta base del brazo para que no esté restringida a un desplazamiento angular de 180° , pues es la única limitante de movimiento más importante del brazo. Aunque esta limitante, fue superada por el desplazamiento de la plataforma móvil, el cambio de junta permitiría que el brazo tuviera muchas más aplicaciones futuras. Se propone colocar un motor de CD en lugar del servomotor.
- El diseño de este robot, permite agregar el último grado de libertad, para la muñeca, pues hace posible que el último eslabón del brazo se cambie, esta es una tarea simple, que además, ampliaría el número de aplicaciones para el brazo, sobre todo, podría manipular objetos en sitios de difícil acceso.

Para realizar una tarea sencilla, como la que se presenta en este trabajo, la cinemática de ambas partes es necesaria y suficiente, pues los movimientos ejecutados por cada uno son satisfactorios. No obstante, en la cinemática inversa del brazo manipulador se conoce, cómo en un futuro, el objeto a manipular será tomado. Sería muy importante darle la posibilidad al sistema de saber cómo debe de sujetar al objeto.

También cabe la posibilidad de implementar este mismo sistema considerando la dinámica de ambas partes, lo que hace al sistema más complejo, sin embargo, determinaría la necesidad de emplear el modelo dinámico para ciertas tareas, como la presente.

El mejor avance que podría lograrse en futuro sería respecto a la evolución de un sistema de visión. Desde la perspectiva del autor, sería mucho mejor si el sistema de visión se montará en el manipulador móvil, de tal manera que funcionara como sus ojos y formara su propio camino. Esta tarea requeriría más tiempo dedicado al algoritmo que a su vez, podría adaptarse, asimismo, la planeación de movimientos se modificaría. Lo anterior, plantea la necesidad de agregar sensores al robot. De hecho, se podrían agregar sensores a cada parte del MM, por ejemplo:

- Sensores laser para la planeación de movimiento de la plataforma móvil, así como sensores de proximidad para poder evadir obstáculos y trazar un mapa de su entorno.
- Para el brazo manipulador puede incorporarse un sensor de proximidad para identificar la distancia a la que se halla el objeto a manipular y pueda corregir el movimiento, si el objeto, estuviera lejos. Cabe mencionar que al realizar este experimento llegó a ocurrir que el MM llegó al *docking* pero por alguna razón no identificada no podía sujetar al objeto. En algunas ocasiones se encontraba muy lejos de él a pesar de que el sistema de visión indicaba lo contrario, por ello surge la necesidad de tener algún sensor de proximidad al objeto. Una propuesta, es colocar un imán a los objetos y en *gripper* un sensor magnético así, si el programa determina que se encuentra cerca de recoger el objeto, tenga la posibilidad de comprobar que se encuentre cerca y corrija la posición, esto, de ser necesario. Otra propuesta diferente es la que se implementa en MM comerciales, y consiste en colocar una cámara en el efector final que compruebe que el objeto está cerca y distinga la forma en que se debe recoger.
- Con la finalidad de comprobar que la señal mandada a los servomotores representa la posición calculada, se tendrían que colocar *encoders* de posición o en su defecto potenciómetros lineales en sus juntas para que se pueda comprobar la posición correcta, o tenga que corregirse.

Obviamente todas las propuestas para trabajos futuros, indican una mayor cantidad de procesamiento, por lo que se debe de hacer que el código que forma el experimento se optimice, a manera de lograr un mejor desempeño.

Para mejorar el desempeño del manipulador, al realizar la operación *P&P*, es muy necesario realizar la trayectoria con el tiempo real de ejecución, así que habría que revisar o descubrir una forma de iniciar un cronómetro en el momento en el que el brazo manipulador tenga que recoger o colocar el objeto, y así le sea

permitido un movimiento más natural y continuo para que se pueda definir el tiempo en que se desea realizar.

La técnica de campos potenciales es, en esencia, utilizada para llegar a una meta evadiendo obstáculos, por lo que sería muy importante en futuros trabajos agregar objetos a evadir. Consolidando la importancia del trabajo, la tarea se acercaría más a una situación real en un entorno real no estructurado. Para lograrlo, hay que cambiar el sistema de visión y de aquí se deriva la propuesta de agregar sensores de proximidad para que, cualquier objeto cercano al MM, se evada. El espacio de trabajo en el que se llevó a cabo este experimento fue muy reducido, por ello, cabría la posibilidad de modificar el tamaño del robot, para que tuviera mayor libertad de movimiento en el entorno de trabajo que sea.

Respecto al algoritmo, habrá que hacerle modificaciones acordes a lo que requieran las necesidades del futuro, tales como, cambiar el sistema de visión por uno a bordo del robot. Si se mantiene la forma en que se adquieren las posiciones y orientaciones de los objetos y del MM, las modificaciones se tendrían que hacer de acuerdo a la tarea a desempeñar, siendo éstas menores a comparación de las primeras. Es decir, este algoritmo sirve como un esqueleto para cualquier otra tarea o experimento, sin perder validez al implementarse en otro experimento. Sirve como referencia para realizar más investigación acerca de la coordinación de movimientos y no sólo de un Manipulador Móvil formado por un brazo manipulador montado a una plataforma móvil, también y esto es de fundamental relevancia, para cualquier otro sistema híbrido.

VIII. APÉNDICES

VIII. 1. VISION POR MEDIO DE REACTIVISION.

VIII. 1. 1. CLIENTE TUIO

Method Summary	
void	acceptMessage (java.util.Date date, com.illposed.osc.OSCMessage message) The OSC callback method where all TUIO messages are received and decoded and where the TUIO event callbacks are dispatched
void	addTuiolListener (TuiolListener listener) Adds the provided TuiolListener to the list of registered TUIO event listeners
void	connect () The TuiolClient starts listening to TUIO messages on the configured UDP port All received TUIO messages are decoded and the resulting TUIO events are broadcasted to all registered TuiolListeners
void	disconnect () The TuiolClient stops listening to TUIO messages on the configured UDP port
TuiolCursor	getTuiolCursor (long s_id) Returns the TuiolCursor corresponding to the provided Session ID or NULL if the Session ID does not refer to an active TuiolCursor
java.util.Vector < TuiolCursor >	getTuiolCursors () Returns a Vector of all currently active TuiolCursors
TuiolObject	getTuiolObject (long s_id) Returns the TuiolObject corresponding to the provided Session ID or NULL if the Session ID does not refer to an active TuiolObject
java.util.Vector < TuiolObject >	getTuiolObjects () Returns a Vector of all currently active TuiolObjects
boolean	isConnected () Returns true if this TuiolClient is currently connected.
void	removeAllTuiolListeners () Removes all TuiolListener from the list of registered TUIO event listeners
void	removeTuiolListener (TuiolListener listener) Removes the provided TuiolListener from the list of registered TUIO event listeners

VIII. 1. 2. CODIGO PARA ADQUISICION DE DATOS POR REACTIVISION EN FUNCIÓN S.

```
function sys=mdlOutputs(t,x,u, ~)

%Función para obtener datos de Reactivision

%r -> Robot (FD 1)
%1 -> Objeto1 (FD 2)
%2 -> Caja_Obj1 (FD 3)
%3 -> Objeto2 (FD 4)
```

```

%4 -> Caja_Obj2 (FD 5)

global xr yr thr x1 y1 th1 x2 y2 th2 x3 y3 th3 x4 y4 th4 f1 f2 f3 f4 f5;

import TUIO.*;
tc=TuioClient();
tc.connect();
for i=1:5%5
    c=tc.getTuioObjects().size();
    if(c>0)
        f1=tc.getTuioObjects().get(0).getSymbolID();
        switch f1
            case 1
                xr=tc.getTuioObjects().get(0).getPosition().getX();
                yr=1-tc.getTuioObjects().get(0).getPosition().getY();
                thr=tc.getTuioObjects().get(0).getAngle();
            case 2
                x1=tc.getTuioObjects().get(0).getPosition().getX();
                y1=1-tc.getTuioObjects().get(0).getPosition().getY();
                th1=tc.getTuioObjects().get(0).getAngle();
            case 3
                x2=tc.getTuioObjects().get(0).getPosition().getX();
                y2=1-tc.getTuioObjects().get(0).getPosition().getY();
                th2=tc.getTuioObjects().get(0).getAngle();
            case 4
                x3=tc.getTuioObjects().get(0).getPosition().getX();
                y3=1-tc.getTuioObjects().get(0).getPosition().getY();
                th3=tc.getTuioObjects().get(0).getAngle();
            case 5
                x4=tc.getTuioObjects().get(0).getPosition().getX();
                y4=1-tc.getTuioObjects().get(0).getPosition().getY();
                th4=tc.getTuioObjects().get(0).getAngle();
            otherwise
                end
        end;
    if(c>1)
        f2=tc.getTuioObjects().get(1).getSymbolID();
        switch f2
            case 1
                xr=tc.getTuioObjects().get(1).getPosition().getX();
                yr=1-tc.getTuioObjects().get(1).getPosition().getY();
                thr=tc.getTuioObjects().get(1).getAngle();
            case 2
                x1=tc.getTuioObjects().get(1).getPosition().getX();
                y1=1-tc.getTuioObjects().get(1).getPosition().getY();
                th1=tc.getTuioObjects().get(1).getAngle();
            case 3
                x2=tc.getTuioObjects().get(1).getPosition().getX();
                y2=1-tc.getTuioObjects().get(1).getPosition().getY();
                th2=tc.getTuioObjects().get(1).getAngle();
            case 4
                x3=tc.getTuioObjects().get(1).getPosition().getX();
                y3=1-tc.getTuioObjects().get(1).getPosition().getY();
                th3=tc.getTuioObjects().get(1).getAngle();
            case 5
                x4=tc.getTuioObjects().get(1).getPosition().getX();
                y4=1-tc.getTuioObjects().get(1).getPosition().getY();
                th4=tc.getTuioObjects().get(1).getAngle();
            otherwise
                end
        end;
    if(c>2)
        f3=tc.getTuioObjects().get(2).getSymbolID();
        switch f3
            case 1
                xr=tc.getTuioObjects().get(2).getPosition().getX();
                yr=1-tc.getTuioObjects().get(2).getPosition().getY();
                thr=tc.getTuioObjects().get(2).getAngle();
            case 2
                x1=tc.getTuioObjects().get(2).getPosition().getX();
                y1=1-tc.getTuioObjects().get(2).getPosition().getY();

```

```

th1=tc.getTuioObjects().get(2).getAngle();
    case 3
x2=tc.getTuioObjects().get(2).getPosition().getX();
y2=1-tc.getTuioObjects().get(2).getPosition().getY();
th2=tc.getTuioObjects().get(2).getAngle();
    case 4
x3=tc.getTuioObjects().get(2).getPosition().getX();
y3=1-tc.getTuioObjects().get(2).getPosition().getY();
th3=tc.getTuioObjects().get(2).getAngle();
    case 5
x4=tc.getTuioObjects().get(2).getPosition().getX();
y4=1-tc.getTuioObjects().get(2).getPosition().getY();
th4=tc.getTuioObjects().get(2).getAngle();
    otherwise
end
end;
if(c>3)
    f4=tc.getTuioObjects().get(3).getSymbolID();
    switch f4
    case 1
xr=tc.getTuioObjects().get(3).getPosition().getX();
yr=1-tc.getTuioObjects().get(3).getPosition().getY();
thr=tc.getTuioObjects().get(3).getAngle();
    case 2
x1=tc.getTuioObjects().get(3).getPosition().getX();
y1=1-tc.getTuioObjects().get(3).getPosition().getY();
th1=tc.getTuioObjects().get(3).getAngle();
    case 3
x2=tc.getTuioObjects().get(3).getPosition().getX();
y2=1-tc.getTuioObjects().get(3).getPosition().getY();
th2=tc.getTuioObjects().get(3).getAngle();
    case 4
x3=tc.getTuioObjects().get(3).getPosition().getX();
y3=1-tc.getTuioObjects().get(3).getPosition().getY();
th3=tc.getTuioObjects().get(3).getAngle();
    case 5
x4=tc.getTuioObjects().get(3).getPosition().getX();
y4=1-tc.getTuioObjects().get(3).getPosition().getY();
th4=tc.getTuioObjects().get(3).getAngle();
    otherwise
end
end;
if(c>4)
    f5=tc.getTuioObjects().get(4).getSymbolID();
    switch f5
    case 1
xr=tc.getTuioObjects().get(4).getPosition().getX();
yr=1-tc.getTuioObjects().get(4).getPosition().getY();
zthr=tc.getTuioObjects().get(4).getAngle();
    case 2
x1=tc.getTuioObjects().get(4).getPosition().getX();
y1=1-tc.getTuioObjects().get(4).getPosition().getY();
th1=tc.getTuioObjects().get(4).getAngle();
    case 3
x2=tc.getTuioObjects().get(4).getPosition().getX();
y2=1-tc.getTuioObjects().get(4).getPosition().getY();
th2=tc.getTuioObjects().get(4).getAngle();
    case 4
x3=tc.getTuioObjects().get(4).getPosition().getX();
y3=1-tc.getTuioObjects().get(4).getPosition().getY();
th3=tc.getTuioObjects().get(4).getAngle();
    case 5
x4=tc.getTuioObjects().get(4).getPosition().getX();
y4=1-tc.getTuioObjects().get(4).getPosition().getY();
th4=tc.getTuioObjects().get(4).getAngle();
    otherwise
end
end;
    pause(0.02);
end;
tc.disconnect();

```

```
sys = [xr yr thr x1 y1 th1 x2 y2 th2 x3 y3 th3 x4 y4 th4];
```

VIII. 1. 3. AJUSTE DE DATOS DE VISIÓN.

```
function sys=mdlOutputs(t,x,u, ~)

global xr yr thr x1 y1 th1 x2 y2 th2 x3 y3 th3 x4 y4 th4 xrr yrr thrr x11 y11 th11 x22 y22
th22 x33 y33 th33 x44 y44 th44;

xr=u(1);
yr=u(2);
thr=u(3);
x1=u(4);
y1=u(5);
th1=u(6);
x2=u(7);
y2=u(8);
th2=u(9);
x3=u(10);
y3=u(11);
th3=u(12);
x4=u(13);
y4=u(14);
th4=u(15);

%****Ajuste de ángulos de 0° a 360° y calculo respecto a horizontal****
thrr=thr+pi/2;
th11=th1+pi/2;
th22=th2+pi/2;
th33=th3+pi/2;
th44=th4+pi/2;

if(thrr>2*pi)
    thrr=thrr-2*pi;
end;
if(th11>2*pi)
    th11=th11-2*pi;
end;
if(th22>2*pi)
    th22=th22-2*pi;
end;
if(th33>2*pi)
    th33=th33-2*pi;
end;
if(th44>2*pi)
    th44=th44-2*pi;
end;
%*****Proyecciones de la posicion del manipulador movil*****
a=12;
A=238;
x_centro=97.75;
y_centro=73.5;
x_MM=xr*195.5;
y_MM=yr*147;
aux1=y_MM-y_centro;
aux2=x_MM-x_centro;

gama=atan2(aux1,aux2);
x_Rmag=norm([aux1 aux2]);
c_g=cos(gama);

x_p=A/(A-a)*x_Rmag;
x_MM_p=x_Rmag*c_g;
x_proy=x_p*c_g;
y_robot=y_centro-x_proy*(y_centro-y_MM)/x_MM_p;
x_robot_r=x_centro+x_proy;
%*****
```

```

sr=sin(thrr);
cr=cos(thrr);
s1=sin(th11);
c1=cos(th11);
s2=sin(th22);
c2=cos(th22);
s3=sin(th33);
c3=cos(th33);
s4=sin(th44);
c4=cos(th44);

***Traslado de Posiciones: de centro de fiducial al centro del objeto***
xrr=x_robot_r+29.3303*cr;
yrr=y_robot+29.3303*sr;

x11=x1*195.5-18*c1;
y11=y1*147-18*s1;
x22=x2*195.5-18*c2;
y22=y2*147-18*s2;
x33=x3*195.5-18*c3;
y33=y3*147-18*s3;
x44=x4*195.5-18*c4;
y44=y4*147-18*s4;
%*****
sys = [xrr yrr thrr x11 y11 th11 x22 y22 th22 x33 y33 th33 x44 y44 th44];

```

VIII. 2. CÓDIGO DE ARDUINO.

```

#include <Wire.h>// Incluir librerías de I2C
#include <Servo.h>//Incluir librerías para controlar servos

#define md25address 0x58
#define speed1 0x00
#define speed2 0x01
#define cmdByte 0x10

//Valores iniciales de juntas de brazo
int b1,b2,h1,h2,c1,c2,m1,m2,gm1,gm2;

int vel_i=128;
int vel_d=128;
int bas=map(90,0,180,530,2360);
int hombr=map(35,0,180,530,2360);
int cod=map(71,0,180,530,2360);
int mune=map(164,0,180,530,2360);
int gmune=544;
int acmune=0;
//
//Configuracion de servomotores
Servo Base;
Servo Hombro;
Servo Hombro2;
Servo Codo;
Servo Muneca;
Servo Gmuneca;
Servo ACmuneca;
//
void setup()

```

```

{
//Configuracion de ancho de pulso de servomotores
Base.attach(3,530,2360);
Hombro.attach(5,530,2360);
Hombro2.attach(6,500,2280);
Codo.attach(9,530,2360);
Muneca.attach(10,530,2360);
Gmuneca.attach(11);
ACmuneca.attach(12);
//
Wire.begin();
Serial.begin(115200);
encoderReset();
}
void loop()
{
  if(Serial.available(>12)//Comprobar que haya datos en bufer
  {
//Recibir datos
    b1=Serial.read();
    b2=Serial.read();
    h1=Serial.read();
    h2=Serial.read();
    c1=Serial.read();
    c2=Serial.read();
    m1=Serial.read();
    m2=Serial.read();
    gm1=Serial.read();
    gm2=Serial.read();
    acmune=Serial.read();

    vel_d=Serial.read();
    vel_i=Serial.read();
//
//Enviar datos
    bas=pwm(b1,b2);
    hombr=pwm(h1,h2);
    cod=pwm(c1,c2);
    mune=pwm(m1,m2);
    gmune=pwm(gm1,gm2);

    juntas_manı(bas,hombr,cod,mune,gmune,acmune);
    vel_llantas(vel_d,vel_i);
  }
//
  else
  {
    bas=bas;
    hombr=hombr;
    cod=cod;
    mune=mune;
    gmune=gmune;
    acmune=acmune;
  }
}

```

```

    vel_d=vel_d;
    vel_i=vel_i;

    juntas_mani(bas,hombr,cod,mune,gmune,acmune);
    vel_llantas(vel_d,vel_i);
}
delay(20);
}
//Reiniciar encoder de motores EMG30
void encoderReset()
{
    Wire.beginTransmission(md25address);
    Wire.send(cmdByte);
    Wire.send(0x20);
    Wire.endTransmission();
}
//
//Enviar datos de velocidad a la tarjeta MD25
void vel_llantas(int vel_d,int vel_i)
{
    Wire.beginTransmission(md25address);
    Wire.send(speed1);
    Wire.send(vel_d);
    Wire.endTransmission();

    Wire.beginTransmission(md25address);
    Wire.send(speed2);
    Wire.send(vel_i);
    Wire.endTransmission();
}
//
//Enviar posiciones angulares a servomotores
void juntas_mani(int base,int hombro,int codo,int muneca,int g_muneca,int ayc_muneca)
{
    Base.writeMicroseconds(base);
    Hombro.writeMicroseconds(hombro);
    int hombro_2=map(hombro,530,2360,0,180);
    Hombro2.write(180-hombro_2);
    Codo.writeMicroseconds(codo);

    Muneca.writeMicroseconds(muneca);
    Gmuneca.writeMicroseconds(g_muneca);
    ACmuneca.write(ayc_muneca);
}
//
//Unir anchos de pulso
int pwm(int x,int y)
{
    int pwm_servo=x*100+y;
    return pwm_servo;
}
//

```

VIII. 3. TARJETA MD25.

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

VIII. 4. CODIGO DE FUNCIONES S.

VIII. 4. 1. ETAPA DE SELECCIÓN.

```
function sys=mdlOutputs(t,x,u, flag)
```

```
global x_r y_r theta_r x_o1 y_o1 theta_o1 x_c1 y_c1 theta_c1 x_o2 y_o2 theta_o2 x_c2 y_c2
theta_c2 p xr yr theta IMM punt sel2;
```

```
x_r=u(1);
y_r=u(2);
theta_r=u(3);
x_o1=u(4);
y_o1=u(5);
theta_o1=u(6);
x_c1=u(7);
y_c1=u(8);
theta_c1=u(9);
x_o2=u(10);
y_o2=u(11);
theta_o2=u(12);
x_c2=u(13);
y_c2=u(14);
theta_c2=u(15);
IPO=u(16);
ILP=u(17);
p=u(18);
sel2=u(19);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Algoritmo de Coordinacion%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if(IPO==0 && ILP==0)
    IMM=0;
    punt=p;
elseif(IPO==0 && ILP==1)
    IMM=1;
    punt=p;
elseif(IPO==2 && ILP==1)
    IMM=0;
    punt=p+1;
elseif(IPO==1 && ILP==1)
```



```

        IMM=1;
        punt=p;
    end

    if(IMM==1)
        sel=mod(p,2);
        if(sel==0)
            IMM=2;
        end
    end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Asignacion del orden de metas del MM%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if(p<=1)
    a=[x_o1-x_r y_o1-y_r];
    b=[x_o2-x_r y_o2-y_r];
    d1=norm(a);
    d2=norm(b);

    if(d2>d1)
        sel2=1;
    else
        sel2=2;
    end
end

switch sel2
    case 1
        Vptos=[x_o1 y_o1 theta_o1;x_c1 y_c1 theta_c1;x_o2 y_o2 theta_o2;x_c2 y_c2 theta_c2];
    case 2
        Vptos=[x_o2 y_o2 theta_o2;x_c2 y_c2 theta_c2;x_o1 y_o1 theta_o1;x_c1 y_c1 theta_c1];
end

if(p~=0 && p<=4)
    xr=Vptos(p,1);
    yr=Vptos(p,2);
    theta=Vptos(p,3);
elseif(p==0)
    xr=Vptos(1,1);
    yr=Vptos(1,2);
    theta=Vptos(1,3);
    punt=1;
else
    IMM=3;
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
sys = [xr yr theta IMM punt sel2];

```

VIII. 4. 2. ETAPA DE PROCESAMIENTO.

VIII. 4. 2. 1. MOVIMIENTO DEL BRAZO MANIPULADOR.

```

function sys=mdlOutputs(t,x,u, flag)

global x_f y_f theta_o x_r y_r theta_r IMM punt x_m y_m z_m phi csi ayc IPO cont;

x_f=u(1);
y_f=u(2);
theta_o=u(3);
x_r=u(4);
y_r=u(5);
theta_r=u(6);
IMM=u(7);
punt=u(8);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
xi=3.5473;

```

```

yi=0;
zi=31.7226;
%*****Parametros de brazo manipulador y cotas finales*****%
zo_f=-9;
zd_f=-5;
h=8.67;
k=0;

if(IMM==0)
%*****Configuracion preferida*****%
x_m=xi;
y_m=yi;
z_m=zi;
phi=0;
csi=0;
ayc=15;
IPO=0;
cont=0;
%*****%
elseif(IMM==1)
%*****Transformacion a coordenadas del brazo manipulador*****%
ct_r=cos(theta_r);
st_r=sin(theta_r);
g=ct_r*(x_f-x_r)+st_r*(y_f-y_r)-h;
l=ct_r*(y_f-y_r)-st_r*(x_f-x_r)-k;
%*****%
IPO=1;
%*****Recogiendo el objeto*****%
if(punt<=20)
[x_m y_m z_m cont]=Trayectoria_mani(xi,yi,zi,g,l,zi,punt);
phi=0;
csi=0;
ayc=15;
elseif(punt>20 && punt<=40)
punt_a=punt-20;
[x_m y_m z_m cont]=Trayectoria_mani(g,l,zi,g,l,zo_f,punt_a);
phi=90*punt_a/20;
csi=Angulo_csi(x_r,y_r,theta_r,theta_o,h,x_f,y_f);%%
ayc=50;
cont=cont+20;
elseif(punt>40)
punt2=punt-40;
if(punt2<=20)
[x_m y_m z_m cont]=Trayectoria_mani(g,l,zo_f,g,l,zi,punt2);
phi=90-90*punt2/20;
csi=Angulo_csi(x_r,y_r,theta_r,theta_o,h,x_f,y_f);
ayc=15;
cont=cont+40;
elseif(punt2>20 && punt2<=40)
punt_a=punt2-20;
[x_m y_m z_m cont]=Trayectoria_mani(g,l,zi,xi,yi,zi,punt_a);
phi=0;
csi=0;
ayc=15;
cont=cont+60;
elseif(punt2>40)
cont=0;
IPO=2;
end
end
%*****Colocando el objeto en su destino*****%
elseif(IMM==2)
ct_r=cos(theta_r);
st_r=sin(theta_r);
g=ct_r*(x_f-x_r)+st_r*(y_f-y_r)-h;
l=ct_r*(y_f-y_r)-st_r*(x_f-x_r)-k;
IPO=1;
if(punt<=20)
[x_m y_m z_m cont]=Trayectoria_mani(xi,yi,zi,g,l,zi,punt);

```

```

    phi=0;
    csi=0;
    ayc=15;
elseif(punt>20 && punt<=40)
    punt_a=punt-20;
    [x_m y_m z_m cont]=Trayectoria_mani(g,l,zi,g,l,zd_f,punt_a);
    phi=90*punt_a/20;
    csi=Angulo_csi(x_r,y_r,theta_r,theta_o,h,x_f,y_f);%%%%
    if(punt==40)
        ayc=50;
    else
        ayc=15;
    end
    cont=cont+20;
elseif(punt>40)
    punt2=punt-40;
    if(punt2<=20)
        [x_m y_m z_m cont]=Trayectoria_mani(g,l,zd_f,xi,yi,zi,punt2);
        phi=90-90*punt2/20;
        csi=0;
        ayc=50;
        cont=cont+40;
    else
        cont=0;
        IPO=2;
    end
end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
else
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Regresando a la configuracion preferida%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    x_m=xi;
    y_m=yi;
    z_m=zi;
    phi=0;
    csi=0;
    ayc=15;
    IPO=2;
    cont=0;
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
sys = [x_m y_m z_m phi csi ayc IPO cont];

function [x_m y_m z_m cont]=Trayectoria_mani(xi,yi,zi,xf,yf,zf,punt)

T=punt/20;
ft=10*T^3-15*T^4+6*T^5;
x_m=xi+(xf-xi)*ft;
y_m=yi+(yf-yi)*ft;
z_m=zi+(zf-zi)*ft;
cont=punt+1;

end

function csi=Angulo_csi(xr,yr,thr,tho,h,xo,yo)

c1=cos(thr);
s1=sin(thr);
ha=xr+h*c1;
ka=yr+h*s1;
eta=atan2(yo-ka,xo-ha);
csi=eta-tho;
if(csi<0)
    csi=pi+csi;
end

end

VIII. 4. 2. 2. CINEMÁTICA INVERSA DEL BRAZO MANIPULADOR.

function sys=mdlOutputs(t,x,u, flag)

```

```

global px py pz phi csi th1 th2 th3 th4 th5;

px=u(1);
py=u(2);
pz=u(3);
phi=u(4);
csi=u(5);
%*****Parametros del brazo manipulador*****%
h=9.3;
k=15;
g=15;
f=11.7;
%*****%
a=cos(phi);
b=sin(phi);

rr=norm([px py]);
x=rr-f*a;
z=pz+f*b;
R=norm([x z-h]);

sin_a=(z-h)/R;
cos_a=x/R;
cos_b=(R^2+g^2-k^2)/(2*g*R);
sin_b=(1-(cos_b)^2)^(1/2);
cos_g=(k^2+g^2-R^2)/(2*g*k);
sin_g=(1-(cos_g)^2)^(1/2);

th1=atan2(py,px);
th2=atan2(sin_a*cos_b+cos_a*sin_b,-(cos_a*cos_b-sin_a*sin_b));
c=cos(th2);
d=sin(th2);
th3=atan2(sin_g,-cos_g);
th4=atan2(a*(sin_g*c-cos_g*d)+b*(cos_g*c+sin_g*d),a*(cos_g*c+sin_g*d)-b*(sin_g*c-cos_g*d));
th5=csi;
sys = [th1 th2 th3 th4 th5];
VIII. 4. 2. 3. ACOPLAMIENTO.

```

```

function sys=mdlOutputs(t,x,u, flag)

global xi yi theta th1 th2 th3 phi xr yr;

xi=u(1);
yi=u(2);
theta=u(3);
th1=u(4);
th2=u(5);
th3=u(6);
phi=u(7);
%*****Parametros del Manipulador Movil*****%
l1=15;
l2=15;
l3=11.7;
xbc=8.67;
ybc=0;
%*****%
c1=cos(pi-th2-th3);
c2=cos(th2);
c3=cos(phi);
c0=cos(th1);
s0=sin(th1);
c=cos(theta);
s=sin(theta);
%*****Acoplamiento*****%
la=l2*c1+l3*c3-l1*c2;

xrc=xbc+la*c0;
yrc=ybc+la*s0;

```

```

xr=xi+xrc*c-yrc*s;
yr=yi+xrc*s+yrc*c;
%*****
sys = [xr yr theta];

```

VIII. 4. 2. 4. LLEGANDO A LA META.

```

function sys=mdlOutputs(t,x,u, flag)

global xi yi thetai xf yf phi caso IMM v w ca ILP x1 y1;

xi=u(1);
yi=u(2);
thetai=u(3);
xf=u(4);
yf=u(5);
phi=u(6);
caso=u(7);
IMM=u(8);
%*****Calculo del pre-docking y el docking*****%
dis_pd=45;
kmm=19;
s=sin(phi);
c=cos(phi);
if(IMM==0)
    if(phi==0||phi==2*pi)
        x1=xf-dis_pd;
        y1=yf;
    elseif(phi==pi/2)
        x1=xf;
        y1=yf-dis_pd;
    elseif(phi==pi)
        x1=xf+dis_pd;
        y1=yf;
    elseif(phi==3/2*pi)
        x1=xf;
        y1=yf+dis_pd;
    else
        y1=yf-dis_pd*s;
        x1=xf-dis_pd*c;
    end
    a=x1-xi;
    b=y1-yi;
    thetag1=atan2(b,a);
    c=xf-xi;
    d=yf-yi;
    thetag2=atan2(d,c);
%*****Asignacion de la meta a la que se dirige el MM*****%
    switch caso
        case 0
            [v,w,ca]=Trayectoria_movil(xi,yi,x1,y1,thetai,thetag1,3);
            ILP=0;
        case 1
            [v,w,ca]=Trayectoria_movil(xi,yi,xf,yf,thetai,thetag2,kmm);%x2--xf y2--yf y kmm
no va
            ILP=0;
        otherwise
            v=0;w=0;ILP=1;
    end
    ca=caso+ca;
else
%*****Plataforma movil detenida*****%
    v=0;
    w=0;
    ILP=1;
    ca=0;
%*****
end

```

```

%*****%
sys = [v w ILP ca x1 y1];

function [vo,wo,cass]=Trayectoria_movil(xi,yi,xf,yf,thetai,thetag,K_mm)
%*****Llegando a la meta por campos potenciales*****%
Kr=10;
vmax=3;
wmax=0.3;

a=xf-xi;
b=yf-yi;
dg=norm([a b]);
thetae=thetag-thetai;
s=sin(thetae);

if(dg<=K_mm)
    vo=0;
    wo=0;
    cass=1;
else
    cass=0;
    if(dg>Kr)
        vo=vmax;
        wo=wmax*s;
    elseif(dg<=Kr)
        vo=(vmax/Kr)*dg;
        wo=wmax*s;
    end
end
%*****%
end

```

VIII. 4. 2. 5. RESTRICCIONES CINEMÁTICAS DE LA PLATAFORMA MÓVIL

```

function sys=mdlOutputs(t,x,u, flag)

global v w theta A B C U Sq Xp;

v=u(1);
w=u(2);
theta=u(3);
d=u(4);
b=u(5);
r=u(6);

U=[v;w];

A=cos(theta);
B=sin(theta);
C=(1/r);

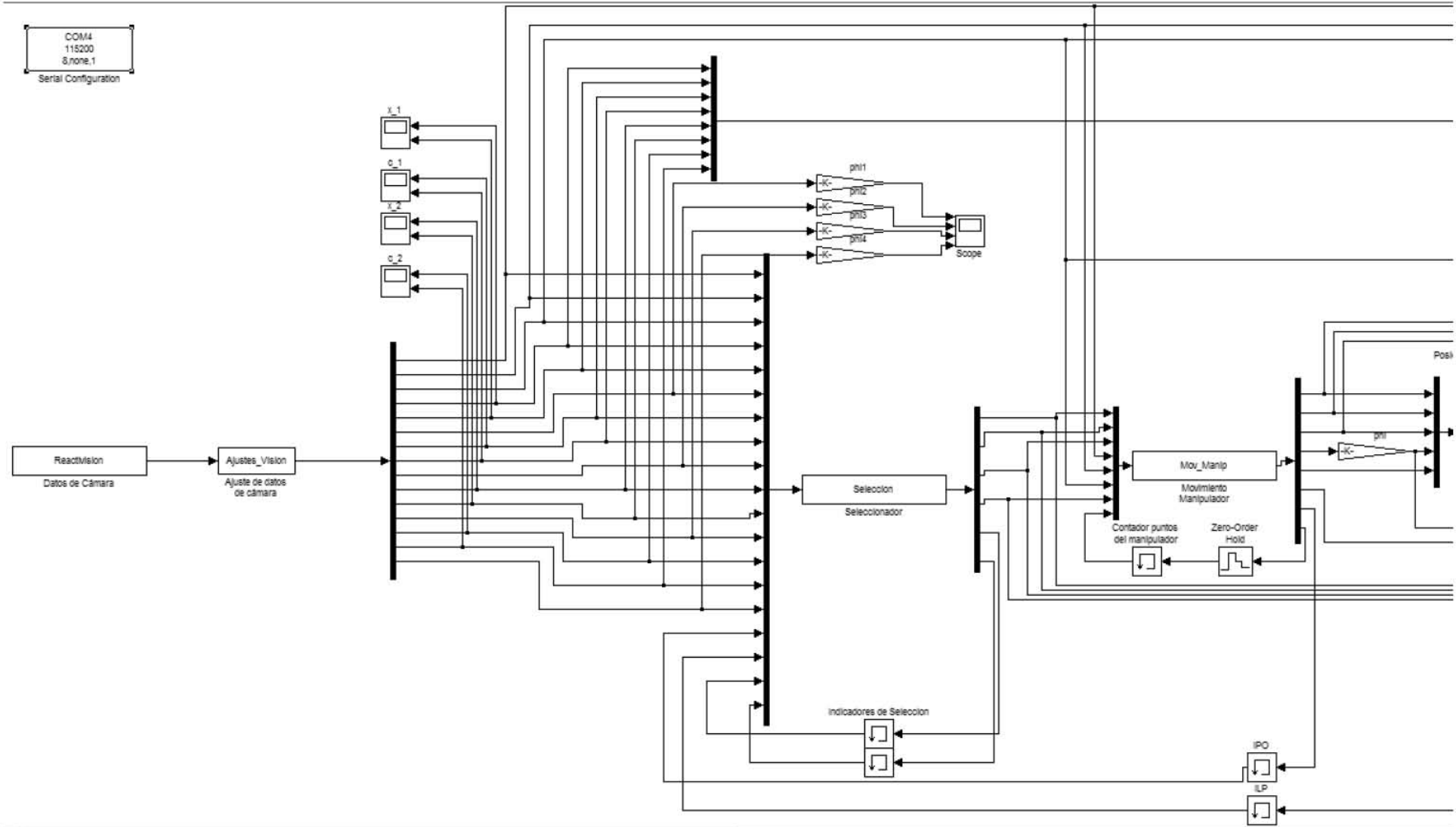
Sq=[A -d*B;B d*A;0 1;C b*C;C -b*C];
Xp=Sq*U;

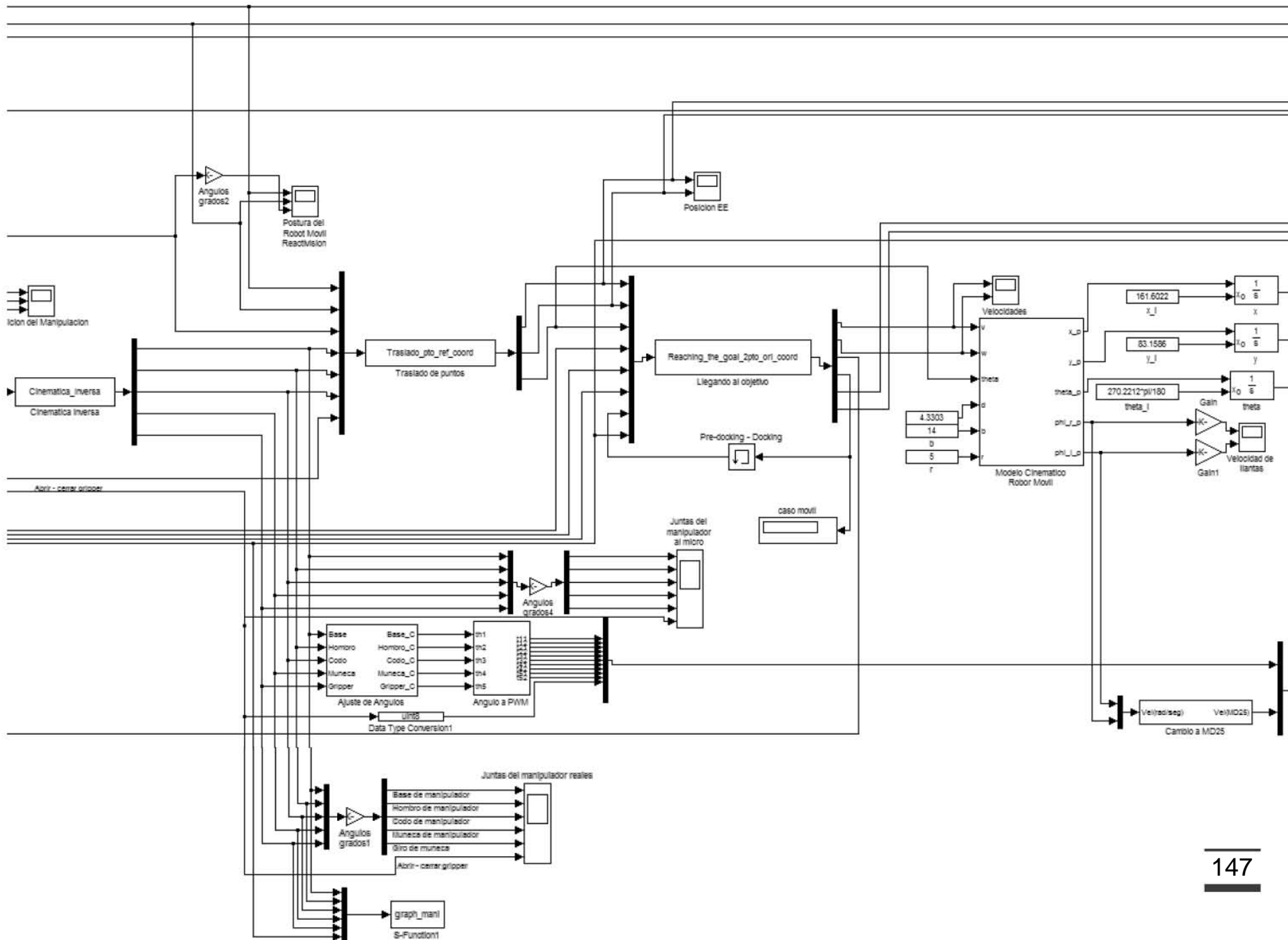
sys = Xp;

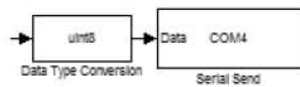
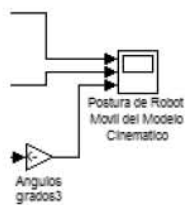
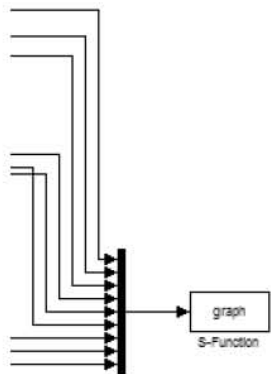
```

VIII. 5. DIAGRAMAS DE BLOQUES DE *SIMULINK*.

VIII. 5. 1. PROGRAMA EN *SIMULINK*.

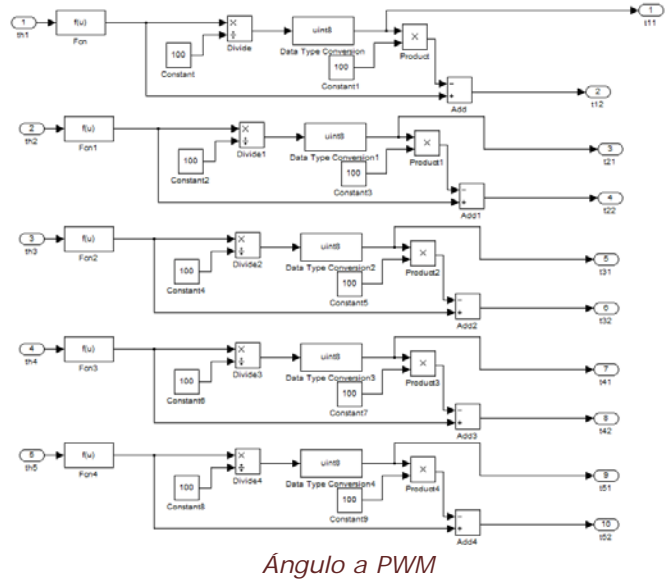
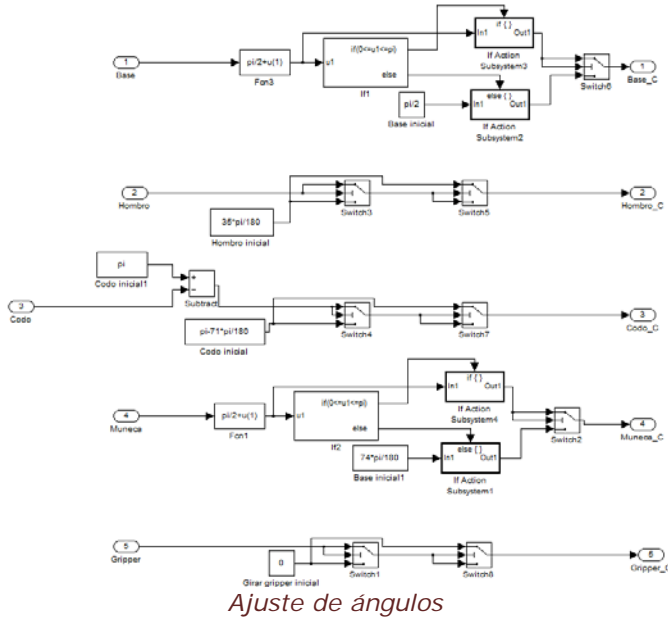




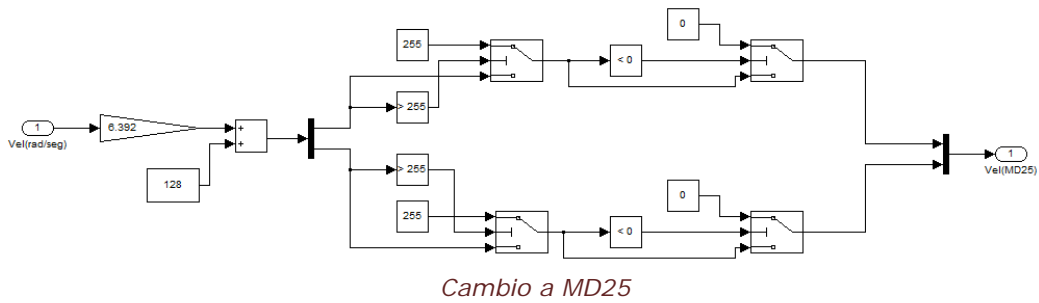


VIII. 5. 2. ETAPA DE EJECUCIÓN.

VIII. 5. 2. 1. AJUSTE DE LOS DATOS DEL BRAZO MANIPULADOR.



VIII. 5. 2. 2. AJUSTE DE LOS DATOS DE LA PLATAFORMA MÓVIL.



IX. REFERENCIAS.

- [1] Yingshu Chen, Libing Liu, Minglu Zhang, Hui Rong; "Study on Coordinated Control and Hardware System of Mobile Manipulator"; 6th World Congress on Intelligent Control and Automation (2006), Dalian, China.
- [2] Gilles Foulon, J. Yves Fourquet, Marc Renaud; "Coordinating mobility and manipulation using nonholonomic mobile manipulators"; Control Engineering Practice 7 (1999), Toulouse, Francia.
- [3] T.I. James Tsay, Y.F. Lai, L. Hsiao; "Material Handling of a Mobile Manipulator Using an Eye in Hand Vision System"; The 2010 IEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan.
- [4] Magnus Egerstedt and Xiaoming Hu; "Coordinated Trajectory Following for Mobile Manipulation"; IEE International Conference on Robotics & Automation (2010), California, USA.
- [5] Chunyan Gao, Minglu Zhang, Lingyu Sun; "Motion Planning And Coordinates Control For Mobile Manipulators"; 9th International Conference on Control, Automation, Robotics and Vision (2006), Tianjin, China.
- [6] Yoshio Yamamoto and Xiaoping Yun; "Coordinating Locomotion and Manipulation of a Mobile Manipulator"; IEEE Transactions on Automatic Control (1994), Philadelphia.
- [7] Paul E. I. Pounds, Daniel R. Bersak and Aaron M. Dollar; "Gasping From the Air: Hovering Capture and Load Stability"; IEE International Conference on Robotics and Automation (2011), Shanghai, China.
- [12] Siddhartha S.S, Dave F, Casey J. H, Dimitry B, Alvaro C, Rosen D, Garrat G, Geoffrey H, James K, Michael V. W; "HERB: a home exploring robotic butler"; Auton Robot (2010), Pittsburgh, USA.
- [13] Mads Hvilshoj and Simon Bogh; "Little Helper – An Mutonomous Industrial Mobile Manipulator Concept"; Intenational Journal of Advance Robotic Systems (2011), Aalborg, Dinamarca.
- [17] J. Angeles; "Fundamentals of Robotic Mechanical Systems: Theory, Methods, and Algorithms"; 3rd edition, Springer (2006), USA.
- [18] K. S. Fu, R. C. González, C. S. G. Lee; "Robótica: Control, detección, visión e inteligencia"; Mc Graw Hill (1989), México.
- [19] J. J. Craig; "Robótica"; Pearson, Prentice Hall, 3a edición (2003), México.
- [20] V. J. G. Villela; "*IV A Unifying Theory on Conventional Wheeled Mobile Robots*. Research on a semiautonomous mobile robot for loosely structured environments focused on transporting mail trolleys" (2006), Reino Unido.
- [21] V. J. G. Villela, et al.; "*A Wheeled Mobile Robot with Obstacle Avoidance Capability*"; SOMIM Ingeniería mecánica tecnología y desarrollo (2004), México.

[25] A. M. Sandoval; “Experimentación en tiempo real sobre deformación reactiva de trayectorias para robótica móvil”; Facultad de Ingeniería, UNAM (2011), México.

[8] http://www.ritsumei.ac.jp/se/~kawamura/research/underwater_e.html

[9] <http://mobilemanipulation.org>

[10] <http://mekabot.com/products/m1-mobile-manipulator/>

[11] <http://www.willowgarage.com/>

[14] <http://www.neobotix-roboter.de/automation-robotics-home.html?&L=1>

[15] <http://www.kuka-robotics.com/germany/en/>

[16] <http://www.youbot-store.com/>

[22] <http://reactivision.sourceforge.net/>

[23] <http://arduino.cc>

[24] <http://www.robot-electronics.co.uk/hm/md25tech.htm>