

# **INSTITUTO POLITÉCNICO NACIONAL**

**ESCUELA SUPERIOR DE INGENIERÍA MECÁNICA Y ELÉCTRICA  
SECCIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN**

## **DISEÑO E IMPLEMENTACIÓN DE ALGORITMOS DE RECUPERACIÓN DE TIEMPOS APLICADOS A UN SISTEMA DE MANUFACTURA FLEXIBLE (FMS)**

**TESIS DOCTORAL  
QUE PARA OBTENER EL GRADO DE  
DOCTOR EN CIENCIAS CON ESPECIALIDAD EN  
INGENIERÍA MECÁNICA**

**PRESENTA:**

**M. en C. JUAN ALEJANDRO FLORES CAMPOS**

**DIRIGIDA POR: DR. LUÍS HÉCTOR HERNÁNDEZ GÓMEZ**

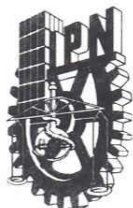


**MÉXICO, D.F.  
NOVIEMBRE 2005**

---

---

---



# INSTITUTO POLITÉCNICO NACIONAL SECRETARIA DE INVESTIGACION Y POSGRADO

## ACTA DE REVISIÓN DE TESIS

En la Ciudad de México, D. F. siendo las 12:00 horas del día 6 del mes de Septiembre del 2005 se reunieron los miembros de la Comisión Revisora de Tesis designada por el Colegio de Profesores de Estudios de Posgrado e Investigación de la E. S. I. M. E. para examinar la tesis de grado titulada:

**“DISEÑO E IMPLEMENTACIÓN DE ALGORITMOS DE RECUPERACIÓN DE TIEMPOS APLICADOS A UN SISTEMA DE MANUFACTURA FLEXIBLE (FMS)”**

Presentada por el alumno:

**FLORES**

**CAMPOS**

**JUAN ALEJANDRO**

Apellido paterno

materno

nombre(s)

Con registro: 

A	0	2	0	3	3	5
---	---	---	---	---	---	---

Aspirante al grado de:

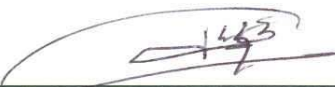
**DOCTOR EN CIENCIAS**

Después de intercambiar opiniones los miembros de la Comisión manifestaron **SU APROBACIÓN DE LA TESIS**, en virtud de que satisface los requisitos señalados por las disposiciones reglamentarias vigentes.

### LA COMISIÓN REVISORA

Director de tesis

  
\_\_\_\_\_  
DR. LUIS HÉCTOR HERNÁNDEZ GÓMEZ

  
\_\_\_\_\_  
DR. GUILLERMO URRIOLAGOITIA CALDERON

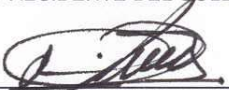
  
\_\_\_\_\_  
DR. PEDRO GUEVARA LÓPEZ

  
\_\_\_\_\_  
DR. JOSÉ DE JESÚS MENDEL JUAREZ

  
\_\_\_\_\_  
DR. ALEXANDER BALANKIN

  
\_\_\_\_\_  
DR. ORLANDO SUSARREY HUERTA

EL PRESIDENTE DEL COLEGIO

  
\_\_\_\_\_  
DR. JAIME ROBLES GARCÍA





**INSTITUTO POLITECNICO NACIONAL**  
COORDINACION GENERAL DE POSGRADO E INVESTIGACION

**CARTA SESION DE DERECHOS**

En la Ciudad de México, Distrito Federal, el día 30 del mes septiembre del año 2005, el (la) que suscribe **JUAN ALEJANDRO FLORES CAMPOS** alumno(a) del Programa de DOCTORADO EN CIENCIAS CON ESPECIALIDAD EN INGENIERÍA MECÁNICA con número de registro A020335, adscrito a la Sección de Estudios de Posgrado e Investigación de la ESIME Unidad Zacatenco, manifiesta que es autor(a) intelectual del presente Trabajo de Tesis bajo la dirección del DR. LUIS HECTOR HERNÁNDEZ GÓMEZ y cede los derechos del trabajo intitulado: "DISEÑO E IMPLEMENTACIÓN DE ALGORITMOS DE RECUPERACIÓN DE TIEMPOS APLICADOS A UN SISTEMA DE MANUFACTURA FLEXIBLE (FMS)", al Instituto Politécnico Nacional para su difusión, con fines académicos y de investigación.

.Los usuarios de la información no deben reproducir el contenido textual, graficas o datos del trabajo sin el permiso expreso del autor y/o director del trabajo. Este puede ser obtenido escribiendo a la siguiente dirección:

**INSTITUTO POLITECNICO NACIONAL**  
**ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA**  
**SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION**

UNIDAD PROFESIONAL "ADOLFO LOPEZ MATEOS", EDIF. 5, 2do. PISO  
COL. LINDAVISTA, 07738 MEXICO D. F., TEL. 57 29 60 00 EXT. 54740.

Si el permiso se otorga, el usuario deberá dar el agradecimiento correspondiente y citar la fuente del mismo.

  
\_\_\_\_\_  
JUAN ALEJANDRO FLORES CAMPOS

Nombre y firma

---

# INDICE GENERAL

ÍNDICE DE FIGURAS Y TABLAS

v

OBJETIVO	xiii
JUSTIFICACIÓN	xiii
RESUMEN	xv
ABSTRACT	xvi
<b>INTRODUCCIÓN</b>	<b>xvii</b>
ELEMENTOS QUE INTEGRAN EL DESARROLLO DEL PROYECTO DE INVESTIGACION	xix
ORGANIZACIÓN DE LA TESIS	xxii
TRABAJOS REALIZADOS EN LA SEPI-ESIME	xxiii
REFERENCIAS	xxv
<b>CAPÍTULO I ESTADO DEL ARTE</b>	<b>1</b>
<b>I.1 RECIENTES AVANCES RELACIONADOS CON ROBOT INTERCEPTORES</b>	<b>1</b>
<b>I.2 SISTEMA DE MANUFACTURA INTEGRADO POR COMPUTADORA (CIM)</b>	<b>15</b>
<b>I.2.1 LAS MÁQUINAS AUTOMÁTICAS</b>	<b>15</b>
<b>I.2.2 SISTEMAS INTERMITENTES</b>	<b>18</b>
<b>I.2.3 APLICACIÓN DE UN CASO REAL. DESCRIPCIÓN DEL SISTEMA</b>	<b>20</b>
<b>I.3 PLANTEAMIENTO DEL CASO DEL ESTUDIO</b>	<b>22</b>
<b>I.4 METODOLOGÍA DE ESTUDIO</b>	<b>23</b>
<b>I.5 CONTRIBUCIONES PRINCIPALES DE LA INVESTIGACIÓN</b>	<b>24</b>
<b>I.6 SUMARIO</b>	<b>25</b>
<b>I.7 REFERENCIAS</b>	<b>26</b>
<b>CAPÍTULO II MARCO TEÓRICO</b>	<b>29</b>
<b>II.1 GENERALIDADES</b>	<b>29</b>
<b>II.2 ALGORITMOS DE INTERCEPCIÓN MÚLTIPLE DE TRAYECTORIAS BIDIMENSIONALES</b>	<b>30</b>
<b>II.3 INTERCEPCIÓN DE OBJETIVOS MÚLTIPLES</b>	<b>31</b>
<b>II.4 ANÁLISIS DEL PROBLEMA DE INTERCEPCIÓN</b>	<b>32</b>
<b>II.4.1 EL FENÓMENO FÍSICO</b>	<b>32</b>
<b>II.4.2 EL FENÓMENO REAL</b>	<b>34</b>
<b>II.5 PLANTEAMIENTO DEL MODELO</b>	<b>36</b>
<b>II.6 SOLUCIÓN AL CASO DE LA TRAYECTORIA DEL TIRO PARABÓLICO</b>	<b>36</b>
<b>II.6.1 ANÁLISIS DE POSICIÓN</b>	<b>36</b>
<b>II.6.2 ANÁLISIS DE VELOCIDAD</b>	<b>39</b>
<b>II.6.3 ANÁLISIS DE ACELERACIÓN</b>	<b>39</b>
<b>II.7 CAMBIO DE BASE</b>	<b>40</b>
<b>II.7.1 ANÁLISIS DE POSICIÓN</b>	<b>40</b>
<b>II.7.2 ANÁLISIS DE VELOCIDAD Y ACELERACIÓN</b>	<b>41</b>
<b>II.8 INTERCEPCIÓN MÚLTIPLE DE DOS PARTÍCULAS LIBRES</b>	<b>41</b>
<b>II.8.1 EL FENÓMENO REAL</b>	<b>41</b>
<b>II.8.2 ANÁLISIS DE POSICIÓN</b>	<b>44</b>
<b>II.8.3 ANÁLISIS DE VELOCIDAD</b>	<b>46</b>
<b>II.8.4 ANÁLISIS DE ACELERACIÓN</b>	<b>47</b>
<b>II.9 CAMBIO DE BASE DOS PARTÍCULAS</b>	<b>47</b>
<b>II.9.1 ANÁLISIS DE POSICIÓN</b>	<b>48</b>
<b>II.9.2 ANÁLISIS DE VELOCIDAD</b>	<b>50</b>
<b>II.9.3 ANÁLISIS DE ACELERACIÓN</b>	<b>51</b>
<b>II.10 GENERALIZACIÓN DE LA INTERCEPCIÓN MÚLTIPLE DEL FENÓMENO DEL FENÓMENO DE TIRO PARABÓLICO</b>	<b>53</b>
<b>II.11 ÁLGEBRA DE QUATERNIONES PARA LA OBTENCIÓN DEL MODELO CINEMÁTICO DEL ROBOT INTERCEPTOR</b>	<b>58</b>
<b>II.11.1 ÁLGEBRA DE QUATERNIONES</b>	<b>59</b>

---

II.11.2 REPRESENTACIÓN PARAMÉTRICA DE ROTACIONES DE UN CUERPO RÍGIDO	61
II.11.3 ISOMORFISMO DE LOS VECTORES DE $\mathbb{R}^3$ AL ESPACIO VECTORIAL Q	62
II.11.4 ROTACIÓN DE UN SISTEMA CARTESIANO DE REFERENCIA	62
II.11.5 PROPIEDADES DE LA TRANSFORMACIÓN	64
II.11.6 POSICIÓN, VELOCIDAD Y ACELERACIÓN DE DOS CUERPOS RÍGIDOS ACOPLADOS	65
II.12 ALGORITMO DE MODELADO DE N CUERPOS RÍGIDOS CON QUATERNIONES	71
II.13 SUMARIO	73
II.14 REFERENCIAS	75
<b>CAPÍTULO III ANÁLISIS NUMÉRICO PREELIMINAR</b>	<b>77</b>
III.1 GENERALIDADES	77
III.2 SIMULADOR DEL FENÓMENO DE INTERCEPCIÓN MÚLTIPLE	78
III.3 SIMULACIÓN COMPUTACIÓN DEL CASO I	80
III.4 SIMULACIÓN COMPUTACIONAL CAMBIO DE BASE DEL CASO I	87
III.5 SIMULACIÓN COMPUTACIONAL INTERCEPCIÓN MÚLTIPLE DE DOS PARTÍCULAS LIBRES	89
III.6 SIMULACIÓN COMPUTACIONAL INTERCEPCIÓN MÚLTIPLE DE DOS PARTÍCULAS CAMBIO DE BASE	91
III.7 CONFIGURACIÓN NO DEFORMADA Y CONFIGURACIÓN INICIO DE INTERCEPCIÓN DEL ROBOT INTERCEPTOR	95
III.7.1 ANÁLISIS DE LOS CASOS I y II PARA LA INTERCEPCIÓN DE UNA PARTÍCULA	96
III.8 INTERCEPCIÓN DE UNA PARTÍCULA LIBRE VÍA ROBOT MANIPULADOR DE 2 GDL	101
III.9 INTERCEPCIÓN DE DOS PARTÍCULAS LIBRES VÍA ROBOT INTERCEPTOR DE 2 GDL	104
III.10 COORDENADAS RELATIVAS DEL FENÓMENO DE INTERCEPCIÓN MÚLTIPLE	115
III.11 DEFINICIÓN DEL ÁREA DE TRABAJO	118
III.12 ESQUEMA DE CONTROL IMPLEMENTADO APLICADO AL ROBOT INTERCEPTOR PARA LA TAREA DE INTERCEPCIÓN	121
III.12.1 EL MODELO DINÁMICO DEL ROBOT INTERCEPTOR	121
III.12.2 CARACTERÍSTICAS DE LAS FORMULACIONES	122
III.13 CONTROL DE POSICIÓN PURA EN COORDENADAS ARTICULARES	126
III.14 CONTROL DE MOVIMIENTO EN COORDENADAS ARTICULARES	127
III.15 CONTROL CON BASE EN EL MODELO PAR-CALCULADO	128
III.16 SIMULACIÓN COMPUTACIONAL DEL CONTROL BASADO EN EL MODELO PAR-CALCULADO IMPLEMENTADO AL ROBOT INTERCEPTOR	131
III.17 SUMARIO	137
III.18 REFERENCIAS	139
<b>CAPÍTULO IV ANÁLISIS EXPERIMENTAL</b>	<b>141</b>
IV.1 GENERALIDADES	141
IV.2 ROBOT MANIPULADOR (ROBOT INTERCEPTOR)	142
IV.3 DISPARADOR DE PARTÍCULAS MÚLTIPLE	145
IV.4 PARÁMETROS DINÁMICOS DEL ROBOT INTERCEPTOR 2 GDL	146
IV.5 REQUERIMIENTOS DE HARDWARE	147
IV.6 DESCRIPCIÓN GENERAL DE LOS SERVOMOTORES DEL ROBOT INTERCEPTOR	148
IV.6.1 SISTEMA DE SERVOPOSICIONAMIENTO	148
IV.7 PONER EN MARCHA EL SERVOSISTEMA	149
IV.7.1 CONTROL POR CORRIENTE (TORQUE)	150
IV.7.2 CONTROL DE VELOCIDAD	150
IV.7.3 CONTROL DE POSICION	151
IV.7.4 OPERACIÓN DEL SISTEMA DE CONTROL	152
IV.7.4.1 CONTROLADOR DE POSICIÓN	152
IV.7.4.2 CONTROLADOR DE VELOCIDAD	153

---

IV.7.4.3 EL CONTROLADOR DE TORQUE Y LA RETROALIMENTACIÓN	154
IV.8 EL SERVO CONTROLADOR	155
IV.9 PARÁMETROS DE LOS MOTORES	160
IV.10 ANÁLISIS DE RESULTADOS DEL SERVOCONTROL	161
IV.10.1 IMPLEMENTACIÓN DE CONTROLADOR P ANALÓGICO PARA EL GRIPPER	161
IV.10.2 ANÁLISIS DE RESULTADOS DEL CONTROLADOR PARA EL GRIPPER	162
IV.11 AMBIENTE DE PROGRAMACIÓN	162
IV.12 CÁLCULO DEL MOVIMIENTO TRAPEZOIDAL	164
IV.12.1 MOVIMIENTO DE LOS SERVO MOTORES	165
IV.12.2 DIAGRAMAS DE VELOCIDAD-TIEMPO	165
IV.13 EL SISTEMA DE VISIÓN	168
IV.13.1 MODO DE USO	168
IV.13.2 COMPONENTES DEL SISTEMA	169
IV.13.3 CAMARA EN MANO	170
IV.13.2 CAMARA FIJA	170
IV.14 DESCRIPCIÓN DEL SISTEMA ROBÓTICO	171
IV.15 MODELO DEL SISTEMA DE VISIÓN	173
IV.15.1 TRANSFORMACION DE COORDENADAS CARTESIANAS A COORDENADAS DE LA CÁMARA	173
IV.15.2 TRANSFORMACIÓN DE PERPECTIVA	174
IV.15.3 MODELO COMLETO DEL SISTEMA DE VISIÓN	174
IV.15.4 PARAMETRIZACIÓN DEL SISTEMA DE VISIÓN	175
IV.15 EL SISTEMA DE VISIÓN	145
IV.13.1 MODO DE USO	145
IV.13.2 COMPONENTES DEL SISTEMA	145
IV.13.3 CÁMARA EN MANO	146
IV.13.4 CÁMARA FIJA	146
IV.14 DESCRIPCIÓN DEL SISTEMA ROBÓTICO	147
IV.15 MODELO DEL SISTEMA DE VISIÓN	150
IV.15.1 TRANSFORMACIÓN DE COORDENADAS CARTESIANAS A COOR..	150
IV.15.2 TRANSFORMACIÓN DE PERSPECTIVA	151
IV.15.3 MODELO COMPLETO DEL SISTEMA DE VISIÓN	151
IV.15.4 PARAMETRIZACIÓN LINEAL DEL SISTEMA DE VISIÓN	152
IV.16 CALIBRACIÓN DEL SISTEMA DE VISIÓN	178
IV.16.1 OPTIMIZACIÓN LINEAL	178
IV.16.2 SOLUCIÓN EN FORMA CERRADA	178
IV.16.3 MÉTODO EN DOS PASOS	179
IV.17 RESULTADOS EXPERIMENTALES DE CALIBRACIÓN DEL SISTEMA DE VISIÓN	180
IV.18 TAREA DEL SISTEMA DE VISIÓN	183
IV.18.1 GENERALIDADES	183
IV.18.2 CAPTURA Y PROCESAMIENTO DE LA IMAGEN	183
IV.18.3 SEGMENTACIÓN	184
IV.18.4 RECONOCIMIENTO	184
IV.18.5 MOMENTOS INVARIANTES	186
IV.18.6 ANÁLISIS DE RASGOS DISTINTIVOS	187
IV.18.7 DETERMINACION DEL CENTRO DE UN CÍRCULO	192
IV.19 RESULTADOS EXPERIMENTALES DE LA IMPLEMENTACION DEL SISTEMA DE VISION	194
IV.20 SUMARIO	196
IV.21 REFERENCIAS	197
<b>CAPÍTULO V APLICACIÓN DE LOS ALGORITMOS DE RECUPERACIÓN DE TIEMPOS AL MODELO CINEMÁTICO DEL ROBOT INDUSTRIAL MITSUBISHI®</b>	<b>201</b>
V.1 GENERALIDADES	201
V.2 PARÁMETROS FÍSICOS DEL ROBOT MITSUBISHI RV1	201

<b>V.3</b>	<b>MODELACIÓN CINEMÁTICA UTILIZANDO QUATERNIONES</b>	<b>205</b>
<b>V.4</b>	<b>APLICACIÓN DE LA SECUENCIA DE ROTACIONES PARA DETERMINAR LA CONFIGURACION NO DEFORMADA</b>	<b>200</b>
<b>V.5</b>	<b>VECTORES QUE DEFINEN LOS ESLABONES Y POSICIÓN DEL ELEMENTO TERMINAL EN LA CONFIGURACIÓN NO DEFORMADA</b>	<b>209</b>
<b>V.6</b>	<b>CONFIGURACION DEFORMADA</b>	<b>212</b>
<b>V.7</b>	<b>APLICACIÓN DE LA SECUENCIA DE ROTACIONES PARA DETERMINAR LA CONFIGURACION DEFORMADA</b>	<b>216</b>
<b>V.8</b>	<b>VECTORES QUE DEFINEN LOS ESLABONES Y POSICIÓN DEL ELEMENTO TERMINAL EN LA CONFIGURACIÓN DEFORMADA</b>	<b>218</b>
<b>V.9</b>	<b>SUMARIO</b>	<b>222</b>
<b>CAPÍTULO VI</b>	<b>ANÁLISIS DE RESULTADOS EXPERIMENTALES DE LOS EVENTOS REALIZA...</b>	<b>223</b>
<b>VI.1</b>	<b>GENERALIDADES</b>	<b>223</b>
<b>VI.2</b>	<b>ANÁLISIS EXPERIMENTAL DE UN CASO PARTÍCULAR DE TIRO PARABÓLICO</b>	<b>223</b>
<b>VI.3</b>	<b>CONDICIONES Y RESTRICCIONES DEL EXPERIMENTO</b>	<b>225</b>
<b>VI.4</b>	<b>ANÁLISIS DE LAS COORDENADAS ARTICULARES DEL ROBOT INTERCEPTOR</b>	<b>229</b>
<b>VI.5</b>	<b>ANÁLISIS DE LA TRAYECTORIA SEGUIDORA TPS</b>	<b>231</b>
<b>VI.6</b>	<b>RESULTADOS EXPERIMENTALES DEL CASO PARTICULAR EN ESTUDIO</b>	<b>233</b>
<b>VI.7</b>	<b>EL GRIPPER DEL ROBOT INTERCEPTOR</b>	<b>243</b>
<b>VI.8</b>	<b>ANÁLISIS DE RESULTADOS DEL CASO PARTICULAR DE TIRO PARABÓLICO</b>	<b>244</b>
<b>VI.9</b>	<b>PARTE EXPERIMENTAL CON LA ESTACIÓN DE MANUFACTURA FLEXIBLE (FMS)</b>	<b>246</b>
<b>VI.10</b>	<b>SECUENCIA DE PASOS DESCRIPTIVOS DE LA PARTE EXPERIMENTAL</b>	<b>248</b>
<b>VI.11</b>	<b>HISTOGRAMAS DE VELOCIDAD Y DEL VECTOR DE COLOR RGB</b>	<b>252</b>
<b>VI.12</b>	<b>PLANTEAMIENTO DEL PROBLEMA PARA LA PARTE EXPERIMENTAL</b>	<b>256</b>
<b>VI.12.1</b>	<b>PLANTEAMIENTO DEL PROBLEMA EXPERIMENTAL</b>	<b>256</b>
<b>VI.12.2</b>	<b>RESTRICCIONES DE TIEMPO</b>	<b>257</b>
<b>VI.12.3</b>	<b>RESTRICCIONES FÍSICAS</b>	<b>257</b>
<b>VI.13</b>	<b>SOLUCIÓN DEL PROBLEMA</b>	<b>258</b>
<b>VI.14</b>	<b>ANÁLISIS DE LA TRAYECTORIA TPS</b>	<b>260</b>
<b>VI.15</b>	<b>COORDENADAS ARTICULARES DEL ROBOT MITSUBISHI®</b>	<b>270</b>
<b>VI.16</b>	<b>RESULTADOS EXPERIMENTALES DEL MOVIMIENTO DEL EFECTOR FINAL DEL ROBOT MITSUBISHI®.</b>	<b>274</b>
<b>VI.17</b>	<b>SUMARIO</b>	<b>268</b>
<b>VI.18</b>	<b>REFERENCIAS</b>	<b>269</b>
	<b>CONCLUSIONES</b>	
	<b>TRABAJOS FUTUROS</b>	
<b>ANEXO A</b>		<b>A1</b>
<b>ANEXO B</b>		<b>B1</b>
<b>ANEXO C</b>		<b>C1</b>

## ÍNDICE DE FIGURAS Y TABLAS

<b>Figura 1</b>	Sistema mecánico leva y seguidor	<b>xix</b>
<b>Figura 2</b>	Elementos que integran el desarrollo del proyecto en forma esquemática	<b>xx</b>
<b>Figura 1.1</b>	Vista esquemática del sistema robot interceptor	<b>1</b>
<b>Figura 1.2</b>	El robot de 7 gdl tiene como efector final una red circular de 16 cm de diámetro	<b>2</b>
<b>Figura 1.3</b>	Robot de 7 gdl y sistema de visión montado en el techo llamado EYE-FAST-GIMBAL	<b>4</b>
<b>Figura 1.4</b>	Comparación de aceleración en robots convencionales	<b>6</b>
<b>Figura 1.5</b>	Vuelo de la pelota en un ambiente incontrolado	<b>7</b>
<b>Figura 1.6</b>	El sistema binocular multi degree of freedom (MDOF)	<b>9</b>
<b>Figura 1.7</b>	Se muestra la técnica híbrida en donde el robot se mueve bajo el control de AIPNG y en un punto B, una técnica convencional de rastreo toma el control (la posición del efector final del robot con el objetivo es minimizado continuamente).	<b>11</b>
<b>Figura 1.8</b>	Clasificación visual instalado sobre una escena industrial	<b>12</b>
<b>Figura 1.9</b>	Arreglo óptico para el sistema de reporte de guía. La luz esparcida del objeto móvil es colectada por un espejo cóncavo y después se realiza filtrado especial, se enfoca en una fotocelda utilizando las ópticas apropiadas.	<b>14</b>
<b>Figura 1.10</b>	PLC Central de un sistema de Manufactura Integrado por Computadora	<b>16</b>
<b>Figura 1.11</b>	Vagón en espera de ser atendido por el módulo de FMS. El robot Mitsubishi® toma la pieza cuando ésta se detiene en la estación	<b>17</b>
<b>Figura 1.12</b>	Módulo de visión artificial	<b>17</b>
<b>Figura 1.13</b>	Módulo de almacén automático	<b>18</b>
<b>Figura 1.14</b>	Sistema de Manufactura Flexible (FMS).	<b>20</b>
<b>Figura 1.15</b>	Sistema de Manufactura Integrado por Computadora. Laboratorio de Mecatrónica UPIITA-IPN.	<b>21</b>
<b>Figura 2.1</b>	Fenómeno físico de intercepción. TPL.- Trayectoria de la partícula libre. TPS.- Trayectoria de la partícula seguidora	<b>33</b>
<b>Figura 2.2</b>	La función TPL está definida sobre el sistema coordenado $(x(t),t)$ , y la función TPS en $(x^*(t),t)$	<b>34</b>
<b>Figura 2.3</b>	La función TPL está definida sobre el sistema coordenado $(Y_1(t),t)$ , y la función TPS en $(Y_1^*(t),t)$	<b>35</b>
<b>Figura 2.4</b>	La función TPL está definida sobre el sistema coordenado $(Y_2(t),t)$ , y la función TPS en $(Y_2^*(t),t)$	<b>35</b>
<b>Figura 2.5</b>	Representación gráfica del cambio de base para el fenómeno de intercepción	<b>39</b>
<b>Figura 2.6</b>	Fenómeno físico de intercepción múltiple de dos partículas; donde TPL1.- es la trayectoria de la partícula libre 1 y TPL2.- la trayectoria de la partícula 2. TPS.- Trayectoria de la partícula seguidora.	<b>42</b>
<b>Figura 2.7</b>	Las funciones TPL1 y TPL2 están definidas sobre el sistema coordenado $(\mathbf{X}(t),t)$ , y la función TPS en $(\mathbf{X}^*(t),t)$ .	<b>43</b>
<b>Figura 2.8</b>	Las funciones TPL1 y TPL2 están definidas sobre el sistema coordenado $(\mathbf{Y}_1(t),t)$ , y la función TPS en $(\mathbf{Y}_1^*(t),t)$ .	<b>43</b>
<b>Figura 2.9</b>	Las funciones TPL1 y TPL2 están definidas sobre el sistema coordenado $(\mathbf{Y}_2(t),t)$ , y la función TPS en $(\mathbf{Y}_2^*(t),t)$ .	<b>44</b>
<b>Figura 2.10</b>	Representación gráfica del cambio de base para el fenómeno de intercepción.	<b>48</b>
<b>Figura 2.11a</b>	Representación gráfica del cambio de base para el fenómeno de intercepción eje X(t)	<b>53</b>
<b>Figura 2.11b</b>	Representación gráfica del cambio de base para el fenómeno de intercepción eje Y(t)	<b>53</b>
<b>Figura 2.11c</b>	Representación gráfica del cambio de base para el fenómeno de intercepción eje X(t)	<b>55</b>
<b>Figura 2.11d</b>	Representación gráfica del cambio de base para el fenómeno de intercepción eje Y(t)	<b>55</b>
<b>Figura 2.11e</b>	Representación gráfica del cambio de base para el fenómeno de intercepción eje Y(t)	<b>56</b>
<b>Figura 2.11f</b>	Ejemplo de localización de los sistemas de coordenadas para TPL1 y TPL2	<b>58</b>



<b>Figura 2.11</b>	Representación de la rotación del sistema de coordenadas $\{e_j\}$ , $j = 1,2,3$ para convertirlo en $\{e'_j\}$ .	<b>63</b>
<b>Figura 2.12</b>	Sistema de referencia. Configuración 1.	<b>64</b>
<b>Figura 2.13</b>	Sistemas de referencia. Configuración 2.	<b>66</b>
<b>Figura 2.14</b>	Sistemas de referencia. Configuración 3.	<b>67</b>
<b>Figura 2.15</b>	Sistema de N cuerpos acoplados	<b>71</b>
<b>Figura 3.1</b>	Simulador para realización de eventos de intercepción múltiple.	<b>78</b>
<b>Figura 3.1a</b>	Primera partícula interceptada en el tiempo 0.6 s	<b>78</b>
<b>Figura 3.1b</b>	Primer partícula interceptada en el tiempo 1.1 s.	<b>79</b>
<b>Figura 3.2a</b>	Trayectorias de la partícula libre y la partícula seguidora, en el instante de tiempo $t = 0.30$ s.	<b>81</b>
<b>Figura 3.2b</b>	Trayectorias de la partícula libre y la partícula seguidora, en el instante de tiempo $t = 0.95$ s.	<b>82</b>
<b>Figura 3.3</b>	Trayectorias de la partícula libre y la partícula seguidora, en el instante de tiempo $t_2 = 1.30$ seg. La posición en el tiempo de intercepción de la partícula libre: $\mathbf{X}(t_2) = 3.727$ y $\mathbf{Y}(t_2) = 1.422$ m.	<b>83</b>
<b>Figura 3.4a</b>	Gráfica posición-tiempo, en el instante de tiempo $t = 0.3$ s. Que corresponde al tiempo de retardo $t_1$ . Partícula Libre.	<b>84</b>
<b>Figura 3.4b</b>	Gráfica velocidad-tiempo, en el instante de tiempo $t = 0.3$ s. Que corresponde al tiempo de retardo $t_1$ . Partícula Libre.	<b>84</b>
<b>Figura 3.4c</b>	Gráfica posición-tiempo, en el instante de tiempo $t = 0.3$ s. Que corresponde al tiempo de retardo $t_1$ . Partícula Seguidora.	<b>84</b>
<b>Figura 3.4d</b>	Gráfica velocidad-tiempo, en el instante de tiempo $t = 0.3$ s. Que corresponde al tiempo de retardo $t_1$ . Partícula Seguidora.	<b>84</b>
<b>Figura 3.4e</b>	Gráfica posición-tiempo, en el instante de tiempo $t = 0.95$ s. Partícula Libre.	<b>85</b>
<b>Figura 3.4f</b>	Gráfica velocidad-tiempo, en el instante de tiempo $t = 0.95$ s. Partícula Libre.	<b>85</b>
<b>Figura 3.4g</b>	Gráfica posición-tiempo, en el instante de tiempo $t = 0.95$ s. Partícula Seguidora.	<b>85</b>
<b>Figura 3.4h</b>	Gráfica velocidad-tiempo, en el instante de tiempo $t = 0.95$ s. Partícula Seguidora.	<b>85</b>
<b>Figura 3.4i</b>	Gráfica posición-tiempo, en el instante de tiempo $t = 1.30$ s. Que corresponde al tiempo de intercepción $t_2$ . Partícula Libre.	<b>86</b>
<b>Figura 3.4j</b>	Gráfica velocidad-tiempo, en el instante de tiempo $t = 1.30$ s. Que corresponde al tiempo de intercepción $t_2$ . Partícula Libre.	<b>86</b>
<b>Figura 3.4k</b>	Gráfica posición-tiempo, en el instante de tiempo $t = 1.30$ s. Que corresponde al tiempo de intercepción $t_2$ . Partícula Seguidora.	<b>86</b>
<b>Figura 3.4l</b>	Gráfica velocidad-tiempo, en el instante de tiempo $t = 1.30$ s. Que corresponde al tiempo de intercepción $t_2$ . Partícula Seguidora.	<b>86</b>
<b>Figura 3.5</b>	Trayectorias de la partícula libre y la partícula seguidora, en el tiempo $t_2 = 1.25$ s. La posición en el tiempo de intercepción es $\mathbf{X}(t_2) = 3.136$ y $\mathbf{Y}(t_2) = 0.507$ m. y $\mathbf{a} = 2$ m.	<b>87</b>
<b>Figura 3.6a</b>	Gráfica posición-tiempo en el instante de tiempo $t = 0.3$ s. Que corresponde al tiempo de retardo $t_1$ . Partícula libre.	<b>88</b>
<b>Figura 3.6b</b>	Gráfica velocidad-tiempo en el instante de tiempo $t = 0.3$ s. Que corresponde al tiempo de retardo $t_1$ . Partícula libre.	<b>88</b>
<b>Figura 3.6c</b>	Gráfica posición-tiempo en el instante de tiempo $t = 0.3$ s. Que corresponde al tiempo de retardo $t_1$ . Partícula seguidora.	<b>88</b>
<b>Figura 3.6d</b>	Gráfica velocidad-tiempo en el instante de tiempo $t = 0.3$ s. Que corresponde al tiempo de retardo $t_1$ . Partícula seguidora.	<b>88</b>
<b>Figura 3.7</b>	La trayectoria TPL1 es interceptada primero por la trayectoria TPS, después intercepta la trayectoria TPL2.	<b>89</b>
<b>Figura 3.8a</b>	Gráfica posición-tiempo y velocidad-tiempo en el instante de tiempo $t = 0.3$ s. Que corresponde al tiempo de retardo $t_1$ . Partícula libre.	<b>90</b>
<b>Figura 3.8b</b>	Gráfica posición-tiempo y velocidad tiempo en el tiempo $t = 0.3$ s. de retardo, Que corresponde al tiempo de retardo $t_1$ . Partícula seguidora.	<b>90</b>

<b>Figura 3.8c</b>	Gráfica posición-tiempo y velocidad-tiempo en el instante de tiempo $t = 0.6$ s. Que corresponde al tiempo de intercepción 1, tiempo $t_2$ . Partícula libre.	<b>90</b>
<b>Figura 3.8d</b>	Gráfica posición-tiempo y velocidad tiempo en el tiempo $t = 0.6$ s. Que corresponde al tiempo de intercepción 1, tiempo $t_2$ . Partícula seguidora	<b>90</b>
<b>Figura 3.9</b>	Trayectorias TPL1 y TPL2 en el tiempo $t = 0.3$ s, tiempo de retardo $t_1$ .	<b>92</b>
<b>Figura 3.10</b>	Trayectorias TPL1 y TPL2 en el tiempo $t = 0.6$ s, tiempo de intercepción $t_2$ .	<b>93</b>
<b>Figura 3.11</b>	Trayectorias TPL1 y TPL2 en el tiempo $t = 1.1$ s, tiempo de intercepción $t_3$ .	<b>94</b>
<b>Figura 3.12a</b>	Gráfica posición-tiempo y velocidad-tiempo en el instante de tiempo $t = 0.3$ s que corresponde al tiempo de retardo $t_1$ . Partícula libre.	<b>94</b>
<b>Figura 3.12b</b>	Gráfica posición-tiempo y velocidad tiempo en el tiempo $t = 0.3$ s de retardo, que corresponde al tiempo de retardo $t_1$ . Partícula seguidora.	<b>94</b>
<b>Figura 3.12c</b>	Gráfica posición-tiempo y velocidad-tiempo en el instante de tiempo $t = 0.6$ s que corresponde al tiempo de intercepción $t_2$ . Partícula libre.	<b>95</b>
<b>Figura 3.12d</b>	Gráfica posición-tiempo y velocidad-tiempo en el instante de tiempo $t = 0.6$ s que corresponde al tiempo de intercepción $t_2$ . Partícula seguidora.	<b>95</b>
<b>Figura 3.13</b>	Configuración no deformada e inicio de intercepción.	<b>96</b>
<b>Figura 3.14</b>	Soluciones del algoritmo de intercepción para una partícula libre.	<b>98</b>
<b>Figura 3.15</b>	Soluciones del algoritmo de intercepción para una partícula libre.	<b>98</b>
<b>Figura 3.16</b>	Soluciones del algoritmo de intercepción para una partícula libre.	<b>99</b>
<b>Figura 3.17a</b>	Soluciones del algoritmo de intercepción para una partícula libre.	<b>100</b>
<b>Figura 3.17b</b>	Soluciones del algoritmo de intercepción para una partícula libre.	<b>100</b>
<b>Figura 3.18a</b>	Intercepción de la partícula libre usando el caso I para una partícula libre.	<b>101</b>
<b>Figura 3.18b</b>	Intercepción de la partícula libre usando el caso I para una partícula libre.	<b>102</b>
<b>Figura 3.18c</b>	Intercepción de la partícula libre usando el caso I para una partícula libre.	<b>102</b>
<b>Figura 3.18d</b>	Intercepción de la partícula libre usando el caso I para una partícula libre.	<b>103</b>
<b>Figura 3.18e</b>	Intercepción de la partícula libre usando el caso I para una partícula libre.	<b>104</b>
<b>Figura 3.21</b>	Configuración del robot interceptor en el instante de tiempo absoluto $t = 0$ s.	<b>106</b>
<b>Figura 3.22</b>	Intercepción para dos partículas libres vía robot interceptor, tiempo $t = 0.35$ s.	<b>107</b>
<b>Figura 3.23a</b>	Gráficas de la posición y ecuaciones de interpolación de la posición 1 del robot intercepción.	<b>108</b>
<b>Figura 3.23b</b>	Gráficas de la posición y ecuaciones de interpolación de la posición 1 del robot intercepción.	<b>109</b>
<b>Figura 3.24a</b>	Intercepción para dos partículas libres vía robot interceptor, en el tiempo $t = 0.65$ s.	<b>113</b>
<b>Figura 3.24b</b>	Intercepción para dos partículas libres vía robot interceptor, en el tiempo $t = 0.70$ s.	<b>113</b>
<b>Figura 3.24c</b>	Intercepción para dos partículas libres vía robot interceptor, en el tiempo $t = 1.20$ s.	<b>114</b>
<b>Figura 3.24d</b>	Intercepción para dos partículas libres vía robot interceptor, en el tiempo $t = 0.30$ s.	<b>114</b>
<b>Figura 3.24e</b>	Configuración de inicio de intercepción del robot interceptor en el tiempo absoluto $t = 0.0$ s.	<b>115</b>
<b>Figura 3.24f</b>	Intercepción 1 del robot interceptor en el tiempo absoluto $t = 0.7$ s.	<b>116</b>
<b>Figura 3.24g</b>	Intercepción 2 del robot interceptor en el tiempo absoluto $t = 1.4$ s.	<b>117</b>
<b>Figura 3.25</b>	Esquema que ilustra el rango de intercepción del robot intercepción	<b>118</b>
<b>Figura 3.26</b>	Esquema que representa los dos problemas dinámicos.	<b>122</b>
<b>Figura 3.27</b>	Esquema de Control Par-Calculado.	<b>128</b>
<b>Figura 3.28</b>	Esquema de control PID	<b>131</b>
<b>Figura 3.29</b>	Trayectoria generada por el efector final del robot interceptor a la posición de inicio de intercepción. $(-2.023, 0.03753)$ .	<b>132</b>
<b>Figura 3.30</b>	Trayectoria generada por el efector final del robot interceptor a la posición 1, en este caso punto 3. $(-1.576, 1.836)$ .	<b>134</b>
<b>Figura 3.31</b>	Trayectoria generada por el efector final del robot interceptor a la posición 2, en este caso punto 4. $(0.003622, 0.8472)$ .	<b>135</b>
<b>Figura 4.1</b>	Esquema del planteamiento del trabajo experimental	<b>142</b>
<b>Figura 4.2</b>	Esquema de prototipo de robot interceptor de 2 GDL	<b>143</b>
<b>Figura 4.3</b>	Prototipo experimental de robot interceptor de 2 GDL	<b>144</b>

<b>Figura 4.4</b>	Prototipo de disparador Múltiple	146
<b>Figura 4.5</b>	Sistema típico de control de posición en lazo cerrado	148
<b>Figura 4.6</b>	Configuración mínima de un servo sistema	149
<b>Figura 4.7</b>	Estructura de control en el modo de control por corriente (torque)	150
<b>Figura 4.8</b>	Estructura de control en el modo de control de velocidad	151
<b>Figura 4.9</b>	Estructura de control de posición (pulso y dirección)	152
<b>Figura 4.10</b>	Sistema de control de posición	153
<b>Figura 4.11</b>	Sistema de control de velocidad	154
<b>Figura 4.12</b>	Sistema de control por torque	155
<b>Figura 4.13</b>	Perfil típico de velocidad ideal y típico	157
<b>Figura 4.14</b>	Esquema del lazo de control	159
<b>Figura 4.15</b>	Resultados de control de corriente y velocidad, respectivamente.	161
<b>Figura 4.16</b>	Diagrama de bloques del sistema controlado	161
<b>Figura 4.17</b>	Foto de la electrónica del controlador p (proporcional)	161
<b>Figura 4.18</b>	Control proporcional análogo para regular sistema de posición	162
<b>Figura 4.19</b>	Perfil de movimiento trapezoidal	165
<b>Figura 4.20</b>	Perfil de movimiento trapezoidal	166
<b>Figura 4.21</b>	El área del trapecio se transforma en el área de un rectángulo	167
<b>Figura 4.22</b>	Sistema de coordenadas de referencia	172
<b>Figura 4.23</b>	Sistema de visión y sus coordenadas	173
<b>Figura 4.24</b>	Diagrama de bloques del modelo de visión	175
<b>Figura 4.25</b>	Elemento estructurante 4-conexo	184
<b>Figura 4.26</b>	Vecino 8-conexos del punto p.	185
<b>Figura 4.27</b>	Elemento estructurante	187
<b>Figura 4.28</b>	Geometría para determinar la traslación del centro del rectángulo en términos de la posición de una de las esquinas	188
<b>Figura 4.29</b>	Geometría para determinar la traslación del centro del rectángulo en términos de las longitudes $u$ y $v$ .	189
<b>Figura 4.30</b>	Geometría para determinar las longitudes $u$ y $v$ en términos de las áreas calculadas A, B, C, D, E y F.	190
<b>Figura 4.31</b>	Geometría del círculo su centro y una cuerda	192
<b>Figura 4.32</b>	Círculo proyectado sobre los cuatro cuadrantes de la proyección indexada de la imagen	193
<b>Figura 4.33</b>	Imagen original y binarizada con un umbral de 0.25 en una escala de 0 a 1.	194
<b>Figura 4.34</b>	Seguimiento de la posición de los centroides y ángulos de orientación de los objetos	195
<b>Figura 5.1</b>	Representación esquemática del robot Mitsubishi® y sus ángulos de desplazamiento máximos.	202
<b>Figura 5.2</b>	Representación de los sistemas de coordenadas en el robot Mitsubishi®.	202
<b>Figura 5.3</b>	Rotación del tercer eslabón en la configuración no deformada $g_j, j = 1,2,3$	208
<b>Figura 5.4</b>	Rotación del segundo eslabón en la configuración no deformada $f_j, j = 1,2,3$	209
<b>Figura 5.5</b>	Vectores que definen la pinza del robot Mitsubishi®.	210
<b>Figura 5.6</b>	Vector posición del elemento Terminal en la configuración no deformada	210
<b>Figura 5.6a</b>	Posición no deformada (Home) del robot Mitsubishi® Diferentes vistas	212
<b>Figura 5.7</b>	Rotación del quinto eslabón (pinza)	215
<b>Figura 5.8</b>	Rotación del cuarto eslabón	215
<b>Figura 5.9</b>	Rotación del tercer eslabón	215
<b>Figura 5.10</b>	Rotación del segundo eslabón	215
<b>Figura 5.11</b>	Rotación del primer eslabón	216
<b>Figura 5.12</b>	Cinemática Inversa del robot Mitsubishi® siguiendo una línea recta	221
<b>Figura 5.13</b>	Generación de vectores para controlar la orientación de la pinza	222
<b>Figura 6.1a</b>	Esquema general de los parámetros del evento experimental y montaje	224
<b>Figura 6.1b</b>	Montaje del servo control PID para realizar la parte experimental	224
<b>Figura 6.1c</b>	Interfase gráfica de control PID para realizar la parte experimental.	225
<b>Figura 6.2</b>	La trayectoria TPS	230
<b>Figura 6.3</b>	Evolución de las variables articulares en el tiempo del robot interceptor.	231
<b>Figura 6.4</b>	Posición, velocidad y aceleración angular del eslabón 1 del robot interceptor.	

		<b>232</b>
<b>Figura 6.5</b>	Posición, velocidad y aceleración angular del eslabón 2 del robot interceptor.	<b>233</b>
<b>Figura 6.6</b>	Posición, velocidad y aceleración angular del eslabón 3 del robot interceptor.	<b>233</b>
<b>Figura 6.7</b>	Posición TPSx-t, Velocidad VTPSx-t y ATPSx-t componente X.	<b>234</b>
<b>Figura 6.8</b>	Posición TPSy-t, Velocidad VTPSy-t y ATPSy-t componente Y.	<b>235</b>
<b>Figura 6.9</b>	Gráfica de la posición-tiempo de la trayectoria TPS (X-t)	<b>235</b>
<b>Figura 6.10</b>	Gráfica de la posición-tiempo de la trayectoria TPS (Y-t)	<b>235</b>
<b>Figura 6.11</b>	Eslabón 1: $\Delta t= 0.3$ s, $\theta_1 = -48.3458^\circ$ , $\ddot{\theta}_1 = 2148.7^\circ/s^2$ , $\dot{\theta}_1 = -322.3^\circ/s$ .	<b>237</b>
<b>Figura 6.12</b>	Eslabón 2: $\Delta t= 0.3$ s, $\theta_1 = -23.222^\circ$ , $\ddot{\theta}_1 = 1032.1^\circ/s^2$ , $\dot{\theta}_1 = -154.8^\circ/s$ .	<b>237</b>
<b>Figura 6.13</b>	Eslabón 1: Posición #13, $\theta_1 = -48.3458^\circ$ , $\ddot{\theta}_1 = 2148.7^\circ/s^2$ , $\dot{\theta}_1 = -322.3^\circ/s$ .	<b>237</b>
<b>Figura 6.14</b>	Eslabón 2: Posición #13, $\theta_2 = -23.222^\circ$ , $\ddot{\theta}_2 = 1032.1^\circ/s^2$ , $\dot{\theta}_2 = -154.8^\circ/s$ .	<b>237</b>
<b>Figura 6.15</b>	Posición #13, gráficas combinadas de posición y velocidad angular	<b>238</b>
<b>Figura 6.16</b>	Posición #13, errores de seguimiento de velocidad	<b>238</b>
<b>Figura 6.17</b>	Eslabón 1: Posición #14, $\theta_1 = -7.4^\circ$ , $\ddot{\theta}_1 = 74858.3^\circ/s^2$ , $\dot{\theta}_1 = -741.2^\circ/s$ .	<b>238</b>
<b>Figura 6.18</b>	Eslabón 2: Posición #14, $\theta_2 = 59.6^\circ$ , $\ddot{\theta}_2 = 601915.6^\circ/s^2$ , $\dot{\theta}_2 = 3546.8^\circ/s$ .	<b>238</b>
<b>Figura 6.19</b>	Errores de seguimiento de velocidad	<b>239</b>
<b>Figura 6.20</b>	Eslabón 1: Posición #15, $\theta_1 = -8.1^\circ$ , $\ddot{\theta}_1 = 156283.0^\circ/s^2$ , $\dot{\theta}_1 = -1547.4^\circ/s$ .	<b>239</b>
<b>Figura 6.21</b>	Eslabón 2: Posición #15, $\theta_2 = -39.9^\circ$ , $\ddot{\theta}_2 = -1005208.6^\circ/s^2$ , $\dot{\theta}_2 = 8178.6^\circ/s$ .	<b>239</b>
<b>Figura 6.22</b>	Errores de seguimiento de velocidad	<b>239</b>
<b>Figura 6.23</b>	Eslabón 1: Posición #16, $\theta_1 = -3.4^\circ$ , $\ddot{\theta}_1 = -115282.0^\circ/s^2$ , $\dot{\theta}_1 = -1141.4^\circ/s$ .	<b>240</b>
<b>Figura 6.24</b>	Eslabón 2: Posición #16, $\theta_2 = -32.3^\circ$ , $\ddot{\theta}_2 = -77041.8^\circ/s^2$ , $\dot{\theta}_2 = -977.7^\circ/s$ .	<b>240</b>
<b>Figura 6.25</b>	Errores de seguimiento de velocidad	<b>240</b>
<b>Figura 6.26</b>	Eslabón 1: Posición #17, $\theta_1 = -5.6^\circ$ , $\ddot{\theta}_1 = -22992.9^\circ/s^2$ , $\dot{\theta}_1 = -227.7^\circ/s$ .	<b>240</b>
<b>Figura 6.27</b>	Eslabón 2: Posición #17, $\theta_2 = 0.4^\circ$ , $\ddot{\theta}_2 = 330197.2^\circ/s^2$ , $\dot{\theta}_2 = 3863.5^\circ/s$ .	<b>240</b>
<b>Figura 6.28</b>	Errores de seguimiento de velocidad	<b>241</b>
<b>Figura 6.29</b>	Eslabón 1: Posición #18, $\theta_1 = -2.8^\circ$ , $\ddot{\theta}_1 = -28190.9^\circ/s^2$ , $\dot{\theta}_1 = -279.1^\circ/s$ .	<b>241</b>
<b>Figura 6.30</b>	Eslabón 2: Posición #18, $\theta_2 = 15.9^\circ$ , $\ddot{\theta}_2 = 156599.5^\circ/s^2$ , $\dot{\theta}_2 = 1021.4^\circ/s$ .	<b>241</b>
<b>Figura 6.31</b>	Errores de seguimiento de velocidad	<b>241</b>
<b>Figura 6.32</b>	Eslabón 1: Posición #19, $\theta_1 = -13.3^\circ$ , $\ddot{\theta}_1 = -106116.2^\circ/s^2$ , $\dot{\theta}_1 = -1050.655^\circ/s$ .	<b>242</b>
<b>Figura 6.33</b>	Eslabón 2: Posición #19, $\theta_2 = -29.5^\circ$ , $\ddot{\theta}_2 = -458389.5^\circ/s^2$ , $\dot{\theta}_2 = 4992.9^\circ/s$ .	<b>242</b>
<b>Figura 6.34</b>	Errores de seguimiento de velocidad	<b>242</b>
<b>Figura 6.35</b>	Historial de la secuencia de movimiento i=1,6 del eslabón 1.	<b>242</b>
<b>Figura 6.36</b>	Historial de la secuencia de movimiento i=1,6 del eslabón 2.	<b>242</b>
<b>Figura 6.37</b>	Errores de seguimiento de velocidad de la secuencia.	<b>243</b>
<b>Figura 6.38</b>	Historial de la secuencia de la posición angular de los eslabones	<b>243</b>
<b>Figura 6.39</b>	Relación desplazamiento angular-voltaje.	<b>245</b>
<b>Figura 6.40</b>	Secuencia de imágenes del evento particular experimental de intercepción	<b>242</b>
<b>Figura 6.41</b>	Dimensiones de la pieza, vagón y pallet.	<b>246</b>
<b>Figura 6.42a</b>	Dimensiones del sistema integrado visión-robot Mitsubishi®.	<b>246</b>
<b>Figura 6.42b</b>	Dimensiones del robot Mitsubishi®.	<b>247</b>
<b>Figura 6.43</b>	Distancia (d) máxima y mínima	<b>248</b>
<b>Figura 6.44</b>	Variación en la intensidad de luz del medio ambiente sobre la pieza de la Figura 6.43.	<b>249</b>
<b>Figura 6.45</b>	Color de la pieza a analizar	<b>249</b>
<b>Figura 6.46</b>	Imagen original: a) en el espacio HSV, b) Componente Hue, c) Componente Value, d) componente saturación.	<b>250</b>
<b>Figura 6.47</b>	a) Histograma del componente <b>s</b> , b) Resultado de la binarización	<b>251</b>
<b>Figura 6.48</b>	Vector de color RGB=(231,47,139) para el rosa mostrado	<b>251</b>
<b>Figura 6.49</b>	Recorte de la imagen [1 50 160 50].	<b>251</b>
<b>Figura 6.50</b>	a) Piezas originales, b) piezas procesadas por color	<b>252</b>
<b>Figura 6.51</b>	a) Histogramas de velocidad, distancia, y b) vector de color RGB.	<b>253</b>
<b>Figura 6.52</b>	a) Interfase gráfica, b) corrida de medición de velocidad con el sistema de	

	visión de dos piezas moviéndose sobre la banda	254
<b>Figura 6.53</b>	Secuencia de eventos realizados para obtención de histograma de velocidad.	255
<b>Figura 6.54</b>	a) Esquema general del planteamiento del problema, b) montaje.	256
<b>Figura 6.55</b>	Funciones $x_0$ y $x_1$ , los tiempos en los que la velocidad relativa es cero son $\Delta t_1=t_3-t_2$ y $\Delta t_2=t_5-t_4$ .	258
<b>Figura 6.56</b>	Funciones $y_0$ y $y_1$ , los tiempos en los que la velocidad relativa es cero son $\Delta t_1=t_3-t_2$ y $\Delta t_2=t_5-t_4$ .	258
<b>Figura 6.57</b>	Funciones $V_{x0t}$ y $V_{x1t}$ , los tiempos en los que la velocidad relativa es cero son $\Delta t_1=t_3-t_2$ y $\Delta t_2=t_5-t_4$ .	259
<b>Figura 6.58</b>	Funciones $V_{y0t}$ y $V_{y1t}$ , los tiempos en los que la velocidad relativa es cero son $\Delta t_1=t_3-t_2$ y $\Delta t_2=t_5-t_4$ .	259
<b>Figura 6.59</b>	Posición de inicio de intercepción del robot manipulador Mitsubishi®.	261
<b>Figura 6.60</b>	Gráficas posición, velocidad, aceleración en términos de los componentes $X^*(t)$ y $Y^*(t)$ de la trayectoria TPS.	263
<b>Figura 6.61</b>	Secuencia de movimiento de la simulación de la trayectoria TPS, TPL1 y TPL2 el tiempo de discretización es de $\Delta t = 0.1$ s.	264
<b>Figura 6.62</b>	Secuencia de movimiento de la simulación de la trayectoria TPS, TPL1 y TPL2 el tiempo de discretización es de $\Delta t = 0.1$ s.	265
<b>Figura 6.63</b>	Secuencia de la trayectoria TPS en el rango $t_1=3.0$ a $t_5=8.5$ s con $\Delta t=0.5$ s	266
<b>Figura 6.64</b>	Posiciones de la trayectoria TPS	271
<b>Figura 6.65</b>	Evolución de las coordenadas articulares en función del tiempo $\theta_1(t)$ y $\theta_2(t)$ .	271
<b>Figura 6.66</b>	Evolución de las coordenadas articulares en función del tiempo $\theta_3(t)$ y $\theta_4(t)$ .	271
<b>Figura 6.67</b>	Evolución de las coordenadas articulares en función del tiempo $\theta_5(t)$ .	272
<b>Figura 6.68</b>	Velocidades articulares en función del tiempo.	273
<b>Figura 6.69</b>	Valores del movimiento del efector final del robot Mitsubishi®.	274
<b>Figura 6.70</b>	Secuencia imágenes del evento experimental en el sistema (FMS)	276
<b>Figura 6.71</b>	El sistema de visión en operación.	277
<b>Figura 6.72</b>	Origen mecánico del robot posición del punto p (x,y,z,pitch,roll) = (0.0, 682.0, 300.0, 0.0, 0.0).	278
<b>Figura 6.73</b>	Relación de parámetros de rapidez y velocidad del robot industrial.	280

## TABLAS

<b>Tabla 2.1</b>	Algoritmos para determinar la posición velocidad y aceleración de cuerpos rígidos acoplados	71
<b>Tabla 3.1a</b>	Captura de resultados de la cinemática del fenómeno de intercepción de una partícula libre por una partícula seguidora.	82
<b>Tabla 3.1b</b>	Captura de resultados de la cinemática del fenómeno de intercepción de una partícula libre por una partícula seguidora.	83
<b>Tabla 3.2</b>	Ecuaciones de movimiento del robot interceptor del evento de la Figura 3.22, en términos de coordenadas articulares absolutas	110
<b>Tabla 3.2a</b>	Ecuaciones en coordenadas articulares relativas: del evento mostrado en las Figuras 3.24e, 3.24f, 3.24g.	117
<b>Tabla 3.3</b>	Ganancias $K_p$ , $K_v$ y $K_i$ seleccionadas	132
<b>Tabla 3.4</b>	Gráficas posición y velocidad de los eslabones 1 y 2 del robot interceptor. Inicio de intercepción	133
<b>Tabla 3.5</b>	Gráficas posición y velocidad de los eslabones 1 y 2 del robot interceptor. Posición 1.	134
<b>Tabla 3.6</b>	Gráficas posición y velocidad de los eslabones 1 y 2 del robot interceptor. Posición 2.	136
<b>Tabla 4.1</b>	Notación de los parámetros físicos de la Figura 4.3	145

<b>Tabla 4.2</b>	Parámetros dinámicos del eslabón 1	<b>147</b>
<b>Tabla 4.3</b>	Parámetros dinámicos del eslabón 2	<b>147</b>
<b>Tabla 4.4</b>	Requerimientos del ordenador	<b>147</b>
<b>Tabla 4.5</b>	Componentes del vector de parámetros	<b>177</b>
<b>Tabla 4.6</b>	Vector de parámetros (Sistema de visión libre de distorsión)	<b>178</b>
<b>Tabla 4.7</b>	Resultados experimentales	<b>181</b>
<b>Tabla 4.8</b>	Incrementos al perímetro	<b>185</b>
<b>Tabla 5.1</b>	Parámetros de Denavit-Hartenberg	<b>203</b>
<b>Tabla 5.2</b>	Parámetros físicos del robot	<b>203</b>
<b>Tabla 5.3</b>	Parámetros dinámicos del robot	<b>203</b>
<b>Tabla 5.4</b>	Quaterniones asociados a los eslabones y sus bases locales	<b>206</b>
<b>Tabla 5.5</b>	Ejes de rotación	<b>207</b>
<b>Tabla 6.1</b>	Dimensiones del sistema físico (m)	<b>226</b>
<b>Tabla 6.2</b>	Datos y condiciones iniciales del experimento	<b>226</b>
<b>Tabla 6.3</b>	Secuencia de imágenes del evento experimental. los datos iniciales son: El tiempo de retardo $t_1 = 0.24$ s, El tiempo de intercepción de la TPL1 es $t_2 = 0.26$ s, El tiempo de intercepción de TPL2 es $t_4 = 0.34$ s, y donde $t_3 = t_5 = \Delta t = 0.2$ s.	<b>227</b>
<b>Tabla 6.4</b>	Resultados numéricos de la simulación del evento de intercepción múltiple	<b>230</b>
<b>Tabla 6.5</b>	Aceleraciones angulares máximas permitidas (grados)	<b>236</b>
<b>Tabla 6.6</b>	Velocidades angulares calculadas (grados/s)	<b>236</b>
<b>Tabla 6.7</b>	Aceleraciones con +1% de tolerancia (grados/s <sup>2</sup> )	<b>237</b>
<b>Tabla 6.8</b>	Ganancias posición # 13.	<b>238</b>
<b>Tabla 6.9</b>	Ganancias posición # 14.	<b>239</b>
<b>Tabla 6.10</b>	Ganancias posición # 15.	<b>239</b>
<b>Tabla 6.11</b>	Ganancias posición # 16.	<b>240</b>
<b>Tabla 6.12</b>	Ganancias posición # 17.	<b>241</b>
<b>Tabla 6.13</b>	Ganancias posición # 18.	<b>241</b>
<b>Tabla 6.14</b>	Ganancias posición # 19.	<b>242</b>
<b>Tabla 6.15</b>	Ganancias posiciones: #13 a la # 19.	<b>243</b>
<b>Tabla 6.16</b>	Voltaje para cada posición angular	<b>244</b>
<b>Tabla 6.17</b>	Dimensiones del vagón, pallet y pieza en metros	<b>246</b>
<b>Tabla 6.18</b>	Vectores que definen las posiciones de los componentes del sistema	<b>247</b>
<b>Tabla 6.19</b>	Datos iniciales y definición de tiempos de intercepción	<b>261</b>
<b>Tabla 6.20</b>	Secuencia de movimiento del robot Mitsubishi <sup>®</sup> 5 gdl con la trayectoria TPS, TPL1 y TPL2 con intervalo de tiempo $\Delta t = 0.5$ s.	<b>267</b>
<b>Tabla 6.21</b>	Valores numéricos de las coordenadas articulares absolutas	<b>270</b>
<b>Tabla 6.22</b>	Valores numéricos de las coordenadas articulares relativas	<b>272</b>
<b>Tabla 6.23</b>	Valores numéricos de las velocidades articulares	<b>272</b>
<b>Tabla 6.24</b>	Tiempos requeridos en el evento experimental	<b>278</b>
<b>Tabla 6.25</b>	Especificaciones del robot Mitsubishi <sup>®</sup> .	<b>279</b>

---

## OBJETIVO

**Objetivo Global:** Diseñar, modelar e implementar algoritmos computacionales para la recuperación de tiempos aplicados en un Sistema de Manufactura Flexible (FMS).

**Objetivo Particular 1:** Construir un prototipo de robot-manipulador de dos grados de libertad con el fin de validar la parte teórica. Es decir, los algoritmos diseñados para interceptar dos objetos en pleno vuelo en sus trayectorias conocidas.

**Objetivo Particular 2:** Implementar los algoritmos diseñados en un robot manipulador industrial para que intercepte en forma múltiple dos piezas que se mueven sobre la banda transportadora de un Sistema de Manufactura Flexible (FMS).

**Objetivo Particular 3:** Desarrollar un sistema de visión para determinar los parámetros cinemáticos de piezas u objetos en pleno movimiento sobre la banda transportadora de un Sistema de Manufactura Flexible (FMS)

## JUSTIFICACIÓN

La razón de estudiar el fenómeno de intercepción múltiple de trayectorias vía un robot manipulador para la recuperación de tiempos en un Sistema de Manufactura Flexible(FMS), es debido a que en la actualidad solo se han estudiado técnicas para la intercepción de una sola trayectoria. Lo que se busca demostrar con este trabajo doctoral, es que la recuperación de tiempos en los procesos de palletización puede incrementarse si en una sola secuencia de movimiento del robot manipulador intercepta dos o más piezas u objetos en pleno movimiento sobre una banda transportadora.

La palletización es el proceso en el cual el robot manipulador toma piezas que se mueven sobre una banda transportadora para alimentar a las máquinas de control numérico, para empaquetamiento de productos, realizar alguna actividad de almacenamiento o simplemente mover una pieza de un proceso a otro.

Esta propuesta daría como resultado un aumento en la productividad de este tipo de sistemas de producción en serie, al disminuir el tiempo que se utiliza para que el robot manipulador tome piezas sin que tengan que detenerse sobre la banda transportadora.

El grado de originalidad del trabajo radica principalmente en el desarrollo de algoritmos que intercepten dos o más objetos. Ya que en la literatura actual se han encontrado solo algoritmos que interceptan un solo objeto vía un robot manipulador autónomo, y no existen antecedentes para la intercepción múltiple de trayectorias como se propone en este trabajo. Es aquí, es donde radica la originalidad de este proyecto.

El alcance de las soluciones actuales se enfocan solo en el desarrollo de metodologías

---

para la Intercepción de una sola partícula. Se han utilizado varias técnicas de seguimiento híbridas (combinación de varias técnicas) y esquemas de control para lograr tal fin. Se han trabajado con objetos que vuelan en condiciones ambientales ideales hasta con objetos en los cuales el medio ambiente afecta sus trayectorias, esto se debe a la fricción del aire, al efecto magnus, etc. [17]. En las investigaciones actuales, se han utilizado robots manipuladores de hasta 7 grados de libertad para la intercepción de un objeto. Algunos de estos robots manipuladores utilizan una red en lugar del efector final para atrapar el objeto. No obstante, también se han trabajado con robots que utilizan el efector final (gripper) para atrapar el objeto. En la parte de los sensores para calcular la posición y velocidad de la partícula se han utilizado cámaras de visión montadas en el efector final y afuera del robot manipulador y sensores de posición, velocidad, etc.

Los movimientos de las partículas a interceptar se han realizado utilizando una mesa de control numérico (CNC), tiros parabólicos, caídas libres de objetos, lanzamiento de objetos con la mano, etc. Lo anterior con la finalidad de restringir el movimiento y controlar las variables reales que intervienen en el fenómeno de intercepción.

Con respecto a los controladores de robot manipuladores, se han utilizado esquemas de control como son: control PID, PD, Control adaptable. También se han implementado mecanismos que se mueven con altas aceleraciones y velocidades para capturar el objetos con movimiento rápido.

Las técnicas que se han desarrollado abarcan desde las técnicas de intercepción de proyectiles, misiles y aviones hasta la psicología perceptual que utilizan los humanos en la intercepción de objetos. Los filtros de Kalman para la predicción de las trayectorias de los objetos son frecuentes, y las técnicas de procesamiento de imágenes para reducir los tiempos de seguimiento de los objetos han sido estudiadas. Se han utilizado dos computadoras personales (PC) en la ejecución de la tarea de intercepción de objetos, una de ellas realiza la predicción y el seguimiento de la partícula con un sistema de visión y la otra ejecuta la generación de las trayectorias que debe seguir el efector final del robot manipulador para capturar el objeto, además, de controlar su movimiento.

Las aplicaciones van desde los deportes hasta los robots que capturan satélites descompuestos en el espacio para su reparación, incluyendo la clasificación de piezas en los ambientes industriales.

Diseñando una metodología de estudio para generar dichos algoritmos, se pretende recuperar los tiempos muertos en los módulos de alimentación de los Sistemas de Manufactura Flexible (FMS). El posible impacto de este trabajo se enfoca en mejorar la reducción de los tiempos muertos de espera de los vagones en las estaciones de los módulos.



---

## RESUMEN

En este trabajo se presenta una metodología para la intercepción múltiple de objetos (partículas) que describen trayectorias conocidas vía un robot autónomo. Esta metodología se basa en la técnica conocida en la literatura científica como PPE (Prediction-Planning-Execution), utilizada para interceptar objetos individuales que maniobran lentamente con aceleración constante. Esta metodología propuesta, estudia, analiza y desarrolla las funciones seguidoras para la intercepción múltiple de objetos (partículas) con movimiento libre en condiciones ideales; es decir, sin considerar fuerzas externas, excepto la fuerza debida a la gravedad. Como un caso particular, se estudia y se desarrolla la función seguidora para una partícula libre con movimiento parabólico, luego para dos partículas con velocidades y ángulos de disparo iniciales diferentes. Como caso general, se plantea la metodología para el desarrollo de funciones seguidoras de cualquier tipo de trayectorias conocidas.

La implementación de esta metodología propuesta es validada con ejemplos de simulación numéricos vía un robot autónomo, incluyendo el análisis cinemático de las partículas libres, la partícula seguidora y el robot interceptor.

La implementación experimental consiste en validar los algoritmos de recuperación de tiempos obtenidos con la metodología propuesta con dos casos: en el primero, se implementan los algoritmos en la cinemática inversa de un robot manipulador para interceptar dos esferas en caída libre controlada, utilizando un esquema de control articular. En el segundo caso, se aplican los algoritmos a un robot industrial que opera dentro de un sistema de Manufactura Flexible (FMS), utilizando un sistema de visión para conocer la velocidad y la posición de dos piezas que se mueven sobre la banda transportadora.

La originalidad de este trabajo se enfoca a desarrollar la metodología para interceptar dos o más objetos en movimiento, debido a que actualmente no se ha trabajado con este planteamiento a nivel mundial. La aplicación de esta metodología trae como resultado la recuperación de los tiempos de espera en el proceso de palletización (alimentación de los módulos), aumentado la productividad en los sistemas de Manufactura Flexibe (FMS).

Cabe señalar, que los resultados y contribuciones de este trabajo están dados en simulaciones computacionales y experimentales.

---

## ABSTRACT

In this work a novel methodology is presented for the multiple interception of objects (particles) that describe well-known trajectories via an autonomous robot. This methodology is based on the well-known technique in the scientific literature as PPE (Prediction-Planning-Execution), used to intercept individual objects that maneuver slowly with constant acceleration. The proposed of this methodology is to study, to analyze and develops the functions followers for the multiple interception of objects (particles) with free movement under ideals conditions: without considering external forces except the due force to the gravity. In general, the methodology development can be employed for the functions followers of any type of well-known trajectories. As a particular case of this methodology, we propose the function follower for a particle free with parabolic movement, and the two particles with initial speed and angle of shot different.

The implementation of this methodology is illustrated with numeric simulation via an autonomous robot, including: the cinematic analysis of the free particles, the particle follower and the robot interceptor.

The experimental implementation consists on validating the algorithms of recovery of times obtained with the methodology proposed with two cases: in the first one, the algorithms are implemented in the inverse kinematics of a robot manipulator to intercept two spheres in controlled free fall, using a articulated control. In the second case, the algorithms are applied an industrial robot that operates inside a system of Flexible Factory (FMS), using a system of vision to know the speed and the position of two pieces that move on the conveyor band.

The originality of this work is focused to develop the methodology to intercept two or more objects in movement, because at the moment has not worked with this approach at world level. The application of this methodology brings the recovery of the times of wait as a result in the increased palletization the productivity in the systems of Factory Flexibe (FMS).

It is necessary to indicate that the results and contributions of this work are given in experimental and simulations.

---

# INTRODUCCIÓN

## MECATRÓNICA

Es importante mencionar que la estructuración de este trabajo es multidisciplinario, ya que abarca varias áreas del conocimiento, debido a lo anterior, se utilizará el esquema Mecatrónico para resolver el problema de la intercepción múltiple de piezas vía robot manipulador autónomo, ya que las partes del proyecto pertenecen a varias disciplinas del conocimiento, como son: la mecánica, la electrónica, la computación y algunos aspectos del control. Por lo anterior, la integración del sistema se logrará por medio de un programa en tiempo real no crítico, es decir, los cálculos y decisiones se llevaran a cabo en el tiempo de operación del sistema. Por lo tanto, se presenta una breve descripción de la Mecatrónica.

En la actualidad la concepción de la palabra Mecatrónica ha ido tomando auge, debido a las necesidades de las industrias productoras y manufactureras, donde los conocimientos aislados de Mecánica, electrónica o computación ya no satisfacen la alta exigencia y competitividad de las empresas en el mercado. Los productos que fabrican se vuelven cada vez más sofisticados y complicados en su manufactura y por ende la mano de obra y los conocimientos de los trabajadores también. Por tal motivo, en este trabajo se presenta un panorama global de esta nueva tendencia ideológica y tecnológica llamada Mecatrónica.

La palabra Mecatrónica fue inventada en Japón a finales de los años 60's, por la compañía Yaskawa Electric Company [1]. En los años 70's, la Mecatrónica se consideraba como una combinación de la mecánica y la electrónica; puertas automáticas, máquinas vendedoras, cámaras con autoenfoco, controles, etc. Es decir, consistía en la tecnología servo para lograr un alto desempeño de los productos. Por lo tanto, el concepto era simple pero sentó las bases para el desarrollo de métodos de control avanzados.

En los 80's, se introduce la tecnología informática, los microprocesadores mejoraron aún más el desempeño de los productos mecatrónicos; máquinas de control numérico, robots industriales. Un ejemplo a detalle; son los sistemas de control para automóviles que se integraban con controles de gas, control de vibraciones, control de viaje, sistemas de freno antiderrape, sistemas de bolsas de aire, y muchos más. Éstos fueron algunos de los productos mecatrónicos más populares en aquella época. Después hubo muchas innovaciones tecnológicas, entre ellas la relacionada con el control clásico y los esquemas de control inteligente que se consideraban como control avanzado.

La información tecnológica, que introducía bases de datos en los sistemas mecánicos para mejorar su desempeño, tales como las lavadoras automáticas y otros productos de consumo. Inclusive los métodos sistemáticos de diseño fueron orientados a la Mecatrónica, y pronto los aspectos de modelación e integración de sistemas se volvió más importante, tanto como la ingeniería de diseño y la implementación del control. Cabe mencionar, que la óptica apareció casi al mismo tiempo que la mecatrónica, y

---

entonces un nuevo campo nació: la Optomecatrónica.

En los 90's, el campo de la Mecatrónica introduce la comunicación tecnológica que descentraliza los centros de información a través de redes de información (Internet). Ahora las máquinas pueden controlarse remotamente, como la teleoperación y la telepresencia en los sistemas robóticos. Los sistemas mecatrónicos controlados por redes de computadoras se volvían más populares y a menudo se relacionaban estrechamente con la realidad virtual y el multimedia [6].

En particular, la integración de sistemas tenía otro punto de vista combinando los microsensores con la tecnología de los actuadores, esta integración se define como micro-mecatrónica, trayendo como consecuencia la modificación de los métodos de microfabricación de aquella época. De aquí entonces, que la Mecatrónica ha ido creciendo y abarcando campos como la automatización tecnológica hasta algunos aspectos de la vida en la sociedad.

Existen una infinidad de definiciones para la palabra Mecatrónica y todavía irán aumentando con el tiempo. Sin embargo, la definición para este trabajo es: "La integración sinérgica de la ingeniería mecánica con la electrónica y el control inteligente por computadora, aplicados al diseño y manufactura de productos y procesos industriales" [2].

La palabra sinérgica proviene del vocablo sinergia<sup>1</sup> utilizado en medicina y modificado para usarlo en mecatrónica, se expresa como: "la integración que toma las ventajas de las dos ingenierías para complementar las deficiencias de una con la otra. Es la interacción cooperativa entre los agentes mecánicos y electrónicos para crear un efecto combinado reforzado mayor que la suma de los efectos individuales" [4]. Es importante mencionar que en un sistema mecatrónico, la toma de decisiones se basa en el programa desarrollado para operarlo, pudiendo éste programarse en computadoras o microcontroladores [3]. Ahora bien, el programa debe responder en tiempo real, esta característica lo diferencia del programa convencional, cuyos resultados no sólo son numéricos y lógicamente correctos, deben ser también generados en el tiempo correcto. Un ejemplo excelente se presenta en la figura 1, en donde la Mecatrónica cambia la forma de resolver un problema de tipo mecánico. El sistema presentado está conformado por una leva y un seguidor, utilizado en este caso para moler granos de maíz. El movimiento del seguidor está controlado por la leva impulsada por el motor.

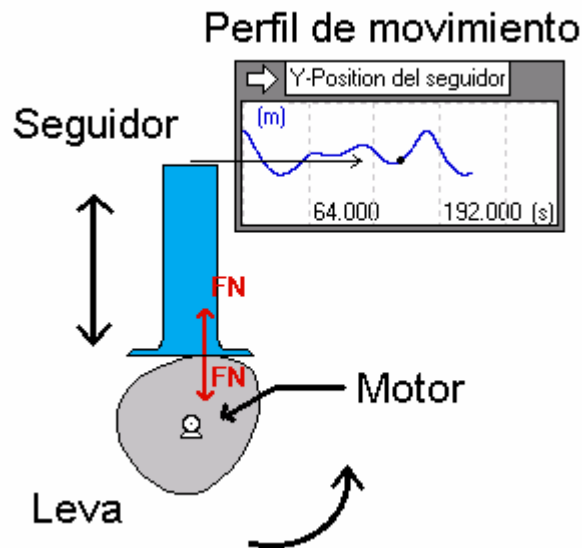
El seguidor genera un movimiento en función del perfil de la leva. Es decir, el movimiento rotacional de la leva se convierte en movimiento lineal y controlado del seguidor. Sin embargo, se requiere de un torque considerable por parte del motor para operar el mecanismo debido a su peso y fuerza de fricción, limitando así su desempeño. Se pierde productividad y el tiempo que se emplea para reemplazar la leva se vuelve importante.

---

<sup>1</sup> En Medicina la palabra Sinergia se entiende como: El efecto combinado de varias sustancias es mayor que la suma de los efectos de las sustancias por separado.

---

Desde el punto de vista mecatrónico, el perfil de movimiento es almacenado en un arreglo de números mediante programación, de tal forma que el desempeño del sistema se ve limitado sólo por las propiedades mecánicas inherentes al motor y al sistema mismo y a la habilidad del control por computadora para almacenar y procesar la información. Cambiar el perfil de movimiento se vuelve una tarea demasiado fácil, sólo basta con introducir la información del nuevo perfil de la leva [3].



**Figura 1** Sistema mecánico Leva y seguidor.

El programa en tiempo real implementado en casi todos los sistemas mecánicos es de seguridad crítica, un mal funcionamiento puede ocasionar serios daños sobre la tarea a realizar y sobre el propio sistema. La palabra clave es sincronía, se vuelve el corazón y el alma del programa en tiempo real [5].

Cabe Mencionar, que los sistemas operativos como Windows® difícilmente pueden realizar una tarea en tiempo real debido a que su estructura es multitarea y sus procesos están orientados al usuario, Por lo tanto, se debe contar con un sistema operativo que trabaje en tiempo real tal como QNX®. Sin embargo, Windows® permite trabajar con tareas que requieren alta prioridad haciendo que el sistema responda en tiempo real no crítico, que en este caso es necesario y suficiente para el desarrollo de este proyecto doctoral.

## **ELEMENTOS QUE INTEGRAN EL DESARROLLO DEL PROYECTO DE INVESTIGACIÓN.**

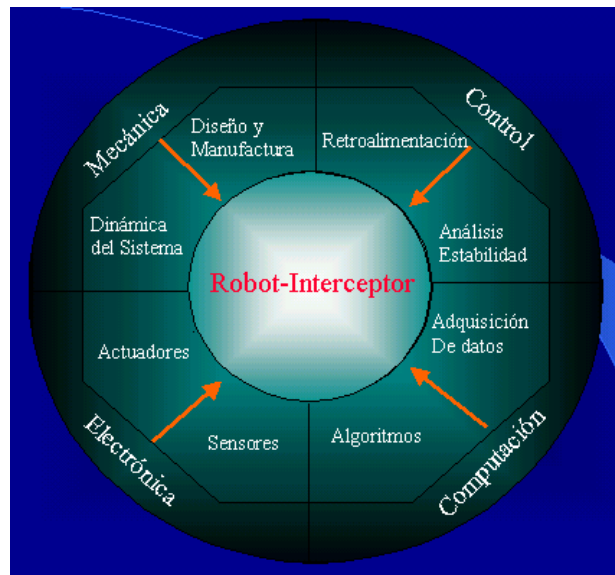
Debido a que el objetivo del presente trabajo se basa en el diseño e implementación de algoritmos de recuperación de tiempos que puedan implementarse a los Sistemas de Manufactura Flexible (FMS), una vez desarrollados los algoritmos de recuperación de tiempos, estos serán validados experimentalmente. Es decir, se implementarán en un

---

prototipo de robot manipulador de dos grados de libertad, y como una aplicación real los algoritmos obtenidos se implementarán también en el controlador del robot manipulador que forma parte de los Sistemas de Manufactura Flexibles (FMS), y se podrán recuperar los tiempos de espera de los vagones en las estaciones de los módulos que integran al CIM. Estos tiempos de espera desaparecen debido a que los vagones no se detienen en las estaciones, sino que el robot manipulador las intercepta en pleno movimiento sobre la banda transportadora. Una vez que el robot captura las piezas, éste alimenta a los módulos de maquinado CNC (Máquinas de Control Numérico) o a algún otro módulo.

Es importante mencionar, que el robot manipulador se llamará robot interceptor, debido a la tarea que realiza.

Por lo tanto, el desarrollo de este proyecto de investigación está relacionado estrechamente con las áreas antes mencionadas. Este Proyecto por su naturaleza abarca las áreas de la Mecánica, Electrónica, Computación y algunos aspectos de control.



**Figura 2** Partes que integran el desarrollo del proyecto en forma esquemática

La figura 2, muestra los elementos que integran el desarrollo de este trabajo de investigación. Cabe mencionar, que el robot interceptor forma parte de un Sistema de Manufactura Flexible (FMS) y que tiene la función de tomar piezas de la banda transportadora para después colocarlas en los centros de maquinado. A continuación se describe brevemente cada uno de los elementos que integran al proyecto:

**Mecánica:** Las áreas de la mecánica que se consideran son:

**a) Diseño y Manufactura:** En esta área de estudio se utilizarán herramientas computacionales para diseñar el robot interceptor, como por ejemplo; el dibujo asistido por computadora (CAD), y para construirlo utilizaremos un programa de Manufactura

---

Asistida por Computadora (CAM), en este caso MasterCam®.

**b) Dinámica del Sistema:** Se utilizará la formulación de Euler-Lagrange para obtener el modelo dinámico del robot interceptor. El cual permitirá aplicar esquemas de control de movimiento del robot interceptor. Y además, se obtendrán los modelos cinemáticos de los robot interceptores apoyándonos en el álgebra de cuaterniones.

**Electrónica:** Las áreas de la electrónica que se consideran son:

**a) Actuadores:** Los actuadores que imprimirán movimiento a los eslabones del robot interceptor serán eléctricos. Es decir, motores de corriente alterna en su modalidad de servomotores controlados por un programa de computadora.

**b) Sensores:** Se Utilizarán sensores de movimiento y velocidad para conocer los parámetros cinemáticos de las piezas a capturar. Estos sensores son elementos electrónicos como por ejemplo: una cámara de visión como sensor externo para conocer los parámetros cinemáticos de las piezas u objetos en movimiento o elementos opto sensibles al movimiento como son los rayos infrarrojos, etc.

**Control:** Los aspectos del control que se emplearán serán los siguientes:

**a) Análisis de estabilidad:** Se realizará el análisis de estabilidad en las zonas de trabajo del robot interceptor para determinar el área adecuada de trabajo de intercepción, y así proponer las ganancias adecuadas para el controlador.

**b) Retroalimentación:** Se utilizarán encoders de cuadratura ópticos para obtener las señales de retroalimentación de posición y velocidad de los motores que mueven los eslabones del robot interceptor. Además, resolvers para generar señales analógicas necesarias para el control.

**Computación:** los aspectos de la computación que se considerarán serán los siguientes:

**a) Adquisición de datos:** La adquisición de datos y el procesamiento de la información se llevará a cabo por medio de una computadora PC. Vía una tarjeta de adquisición de datos que maneja señales analógicas y digitales de entrada y salida. Además, de contar con una tarjeta de adquisición de imágenes, con un programa para la operación del robot como es Work Bench® 5.0, el ARC® 2.0, etc.

**b) Algoritmos:** los algoritmos a diseñar serán programados en lenguajes de programación de alto nivel, y permitirán resolver y realizar el fenómeno de intercepción múltiple de objetos. En esta parte se pueden programar algunas de las técnicas de procesamiento de imágenes en tiempo real. Además, se realizar las simulaciones computacionales del fenómeno de intercepción con software comercial como son: Mathematica® 4.0, Working Model® 4.0. Etc.

---

# ORGANIZACIÓN DE LA TESIS

En el **Capítulo I** se presenta el estado del arte, las aportaciones a nivel internacional como a nivel nacional. Se realiza una descripción de los avances más significativos en el área de la intercepción de trayectorias y se hace énfasis en la aplicación para recuperar los tiempos muertos en los Sistemas de Manufactura Flexible (FMS).

En el **Capítulo II** se plantea el marco teórico del fenómeno de intercepción y se diseñan los algoritmos de recuperación de tiempos. Se inicia el estudio con el fenómeno del tiro parabólico de una partícula y luego se extiende el caso para la intercepción múltiple de dos partículas con sus cambios de base, tanto de la posición del robot, como la posición de disparo de las partículas. Se desarrollan los algoritmos para interceptar objetos con impacto y sin impacto. Posteriormente, se describe en forma breve el álgebra de Cuaterniones para modelar la cinemática del robot interceptor para dos o más cuerpos rígidos. Finalmente, se plantea un algoritmo para modelar cualquier tipo de robot, con juntas rotacionales o juntas prismáticas.

En el **Capítulo III** se realiza el análisis numérico utilizando distintos valores numéricos para identificar los parámetros cinemáticos y mecánicos del robot interceptor. Se desarrolla un simulador para la intercepción múltiple de trayectorias. Además, se acopla el robot interceptor al movimiento de las partículas libres. Por último, se plantea un esquema de control tipo PID con la técnica de par calculado al robot. Con esta información se pretende construir un prototipo de robot interceptor de dos grados de libertad para desarrollar la parte experimental.

En el **Capítulo IV** se diseña y se desarrolla un prototipo de robot interceptor de dos grados de libertad para validar los algoritmos de recuperación de tiempos. Se describe el sistema de visión y el sistema de control de movimiento para el robot interceptor. Se integra el sistema total y se realiza la calibración necesaria para que opere el sistema completo.

En el **Capítulo V** se modela el robot industrial Mitsubishi RV1 usando álgebra de Cuaterniones para implementar la parte experimental dentro del sistema de Manufactura Integrada por Computadora (FMS), se diseña y se realizan pruebas al sistema de visión para el seguimiento de las piezas que se mueven sobre la banda transportadora.

En el **Capítulo VI** se realizan eventos de intercepción y se generan los resultados experimentales, éstos se comparan con los resultados de simulación y se realiza el análisis de resultados y se obtienen las conclusiones. Considerando lo anterior, se proponen trabajos futuros que se convierten en mejoras al sistema de los cuales se derivan nuevos trabajos de investigación.

En los anexos se encuentran los programas desarrollados para presentar las simulaciones, incluyendo las ecuaciones y los algoritmos de recuperación de tiempos, y una breve descripción de los comandos.



---

## TRABAJOS REALIZADOS EN LA SEPI-ESIME .

A nivel nacional existen una infinidad de trabajos relacionados con el diseño y construcción de robots manipuladores, algunos otros trabajos relacionados con los robots móviles. En particular las líneas de investigación en Robótica en la SEPI-ESIME están estrechamente ligadas con la Cinemática Directa e Inversa, la dinámica, el control y la retroalimentación visual.

Los recientes trabajos de investigación desde 1993 abarcan la propuesta de una metodología para el diseño de robots manipuladores [7,12], que marca el inicio del desarrollo del estudio de la Robótica en la Sección. Así como también, el análisis Cinemático y Dinámico de manipuladores Robóticos [8]. El estudio minucioso del esquema de control PID aplicado como regulador a un robot de un grado de libertad, utilizando microcontroladores y una computadora personal PC [13], establece los fundamentos teóricos y prácticos de control aplicado a manipuladores. Lo anterior sin olvidar las consideraciones importantes de control de sistemas electromecánicos en la parte de Modelación y Simulación presentados en [16].

También se ha estudiado la Robótica en el desarrollo de simuladores para realizar control en brazos robóticos [10], e inclusive se han implementado mecanismos con manipuladores para realizar tareas específicas en aplicaciones industriales [9,14,15].

Ahora bien, en el 2003. como una aportación importante se introducen los conceptos de diseño QFD aplicados al diseño de Efectores Finales [9,15], generando las líneas de investigación que se suman a las actuales en la SEPI-ESIME.

Este trabajo de investigación se relaciona con el trabajo de Tesis de Maestría de Merchán Cruz [11], que establece una metodología de generación de trayectorias fuera de línea, utilizando simuladores para un robot de 5 gdl marca Mitsubishi MoveMaster® EV-5 con carga y sin carga, para dar solución a la Cinemática. y Dinámica del robot manipulador. Es importante mencionar, que se aplica el esquema de control PID a un prototipo de manipulador de 2 gdl.

Cabe señalar y enfatizar que los objetivos de la mayoría de los trabajos desarrollados en la SEPI-ESIME están enfocados a eliminar los denominados tiempos muertos derivados de la reprogramación de los robots para la realización de diferentes tareas, en los cuales se detiene la producción generando pérdidas económicas y una baja productividad en los sistemas de Manufactura Integrada por Computadora (CIM) [11,16].

Siguiendo la misma línea de investigación este trabajo ataca el problema desde el punto de vista funcional en donde la reprogramación se realiza en tiempo real con base en algoritmos de recuperación de tiempos. Es decir, en el tiempo de operación del sistema, el robot podrá modificar sus trayectorias para la intercepción de piezas que se mueven sobre la banda transportadora, sin necesidad de detener ningún proceso alterno del sistema. A esta tarea de alimentar a los módulos con piezas que son tomadas desde una banda transportadora vía robot manipulador se le llama palletizar [17,18], el robot

---

interceptor será guiado través de retroalimentación visual. Se puede concluir que este es el primer trabajo doctoral en la SEPI-ESIME en esta área. Y que las aportaciones de los demás trabajos que integran las líneas de investigación en esta sección son de gran interés y beneficio para el desarrollo de esta tesis.

En resumen se han planteado los objetivos a desarrollar en este trabajo de investigación. Además, se ha dado una descripción breve de los elementos que integran el desarrollo de este trabajo, incluyendo el concepto de mecatrónica debido a que el programa será la esencia en el sistema de intercepción múltiple de objetos que se mueven sobre la banda de un sistema de manufactura flexible (FMS), en donde la toma de decisiones y el desarrollo del cálculo de las trayectorias del robot se harán dentro de los intervalos de tiempo establecidos por los algoritmos de recuperación de tiempos. Considerando lo anterior y además de que la modificación será en esencia a través de programación se justifica utilizar el esquema mecatrónico.

---

## REFERENCIAS

- [1] Japan Trademark Registration, no. 946594, Jn. 1972.
- [2] Mechatronics - "What Is It, Why, and How", An Editorial, IEEE/ASME transactions on mechatronics, Vol. 1, No. 1, pags: 1-4, March 1996.
- [3] David M. Auslander, "What is Mechatronics?", IEEE/ASME transactions on mechatronics, Vol. 1, No. 1, pags: 5-9, March 1996.
- [4] Nobuhiro Kyura y Hirosuke Oho, Mechatronic "-An Industrial Perspective", IEEE/ASME transactions on mechatronics, Vol. 1, No. 1, pags:10-15, March 1996.
- [5] Rolf Isermann, Member, IEEE, "Modeling and Desing Methodology for Mechatronic Systems", IEEE/ASME Transactions on Mechatronics, Vol. 1, No. 1, pags: 16-28, Mareh 1996.
- [6] Robin R. Murphy, "Introduction to AI Robotics", Massachusetts of Technology, The MIT Press, 2000, ISBN 0-262-13383-0
- [7] Díaz de León, V. M. "Metodología para el Diseño de un Robot Manipulador Industrial", Tesis de Maestría. SEPI-ESIME-IPN, 1993.
- [8] Ramiro López A, "Análisis Cinemático y Dinámico de un Manipulador Robótico con tres Gados de Libertad", Tesis de Maestría, SEPI-ESIME-IPN, 1993.
- [9] Rojas Garnica, J.C., "Diseño de un mecanismo atornillador para un Brazo Manipulador Robótico", Tesis de Maestría. SEPI-ESIME-IPN, 1997.
- [10] Gonzáles Sánchez, A., "Desarrollo de un simulador para el control de un Brazo Robótica", Tesis de Maestría, SEPI-ESIME-IPN, 1993.
- [11] Merchán Cruz, E. A, "Metodotogía para la Generación de Trayectorias de Manipuladores Robóticos, su Cinemática y Dinámica" , Tesis de Maestría, SEPI-ESIME-IPN, 2000
- [12] B. Solano, T. P, "Diseño de un Manipulador Robótico de cinco grados",Tesis de Maestría. SEPI--ESIME-IPN. 2003.
- [13] V. Sánchez A, T, "Implementación del control PID a un grado de libertad", Tesis de Maestría, SEPI-ESIME-IPN, 2003.
- [14] U. Guerrero H. M, "Diseño de un efector final para pintura", Tesis de Maestría. SEPI-ESIME-IPN,-2003.
- [15] Ramírez G, J, "Diseño de un efector final de Ventosa por Vacío", tesis de Maestría. SEPI-ESIME-IPN, 2003.
- [16] Aguirre, H. "Modelación y Simulación de Sistemas Electromecánicos Bajo Control",Tesis de Maestría, SEPI-ESIME-IPN,1998.
- [17] Russell L. Anderson, A Robot Ping Pong Player: Experiments in Real-time Intelligent Control. Cambridge, MA: The MIT Press, 1988.
- [18] Tien-Chien Chang, Richard A. Wysk, "Computer-Aided Manufacturing", Segunda edición, Prentice Hall International.1998.

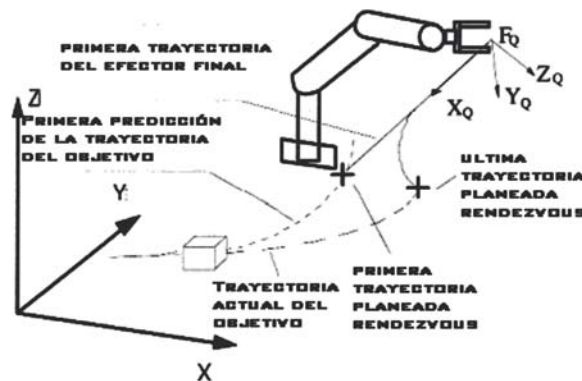
# CAPÍTULO I

## ESTADO DEL ARTE

Se ha revisado la literatura acerca de los robots interceptores de trayectorias, en centros especializados de información. A la fecha, se conocen técnicas de intercepción y se consideran algunos algoritmos de intercepción en el plano y en espacio. A continuación se presentan en breve algunos de los resultados más sobresalientes y las técnicas más utilizadas para resolver la tarea de Intercepción de objetos por medio de robots manipuladores. Así como también, las conclusiones a las que se llegaron y las posibles mejoras.

### I.1 RECIENTES AVANCES RELACIONADOS CON ROBOTS INTERCEPTORES

En [1.1] se propone un nuevo sistema de predicción activa, planeación, predicción y ejecución cuyas siglas son (APPE) aplicado a la intercepción de objetos móviles. La principal característica del sistema propuesto es que tiene la habilidad de interceptar un objeto en cualquier lugar a lo largo de la trayectoria predicha dentro del área de trabajo del robot. En el caso de la industria, donde el movimiento de objetos es predecible durante el tiempo empleado, se puede establecer que esta característica es una mejora significativa sobre una versión APPE anterior, en la cual se puede seleccionar un punto entre los puntos no óptimos considerados para la intercepción.



**Figura 1.1** Vista esquemática del sistema robot interceptor.

El objetivo de este sistema es simple, ya que se trata de mover el robot a la posición antes de que llegue el objeto, tomando en ese instante el control de movimiento. Para este efecto, se emplea un algoritmo de movimiento de rastreo fino, utilizando sensores de proximidad montados sobre el efector final del robot. Así, la necesidad de identificar el movimiento del objeto, minimizando la distancia entre el efector final del robot y el objeto en forma continua, se reduce.

Los resultados que se obtuvieron fueron los siguientes: El sistema experimental puede interceptar objetos con un promedio de error de menos de 10 mm en cada dirección, viajando en trayectorias al azar con velocidades superiores a 45 mm/s y aceleraciones mayores a 200 mm/s<sup>2</sup>. Sin embargo, recordemos que la técnica APPE fue diseñada para llevar el efector final de un robot a una posición preparatoria antes de tocar el objeto. De aquí que el error de 10 mm de magnitud constituye un resultado satisfactorio.

Concluyendo, el nuevo sistema APPE propuesto en este trabajo, complementa a otro sistema APPE previamente sugerido en la literatura para la interceptación de objetos. La llave del éxito del nuevo sistema, es la efectividad de las estrategias para la predicción de la trayectoria-objetivo del objeto y la planeación de trayectoria del robot. Como se menciona anteriormente, el movimiento impredecible del objeto disminuiría la efectibilidad de la interceptación.

En otro orden de ideas, en [1.2] se presenta un sistema para interceptar un objeto esférico en pleno vuelo, empleando un robot-manipulador de 7 grados de libertad. Utilizando componentes off the shelf (cámaras externas al robot) para seguimiento visual. Esto es; el objeto esférico es observado por un sistema de visión estéreo colocada sobre una columna vertical frente al robot manipulador desde la cual se dispara el objeto, comparando cada imagen con otra de referencia. Para este efecto, utilizan seguimiento y predicción de la posición del objetivo. Usando la técnica llamada Filtro de Kalman Extendido (EKF) por sus siglas en inglés que permite considerar el arrastre provocado por el aire sobre la pelota.

**Figura 1.2** El robot de 7 gdl, tiene como efector final una red circular de 16 cm de diámetro para cachar pelotas de 7 cm de diámetro. Algunos datos técnicos son: tiene un peso de 18 kg, tiene 1 m<sup>2</sup> de área radial de trabajo. Una velocidad máxima de 80 °/s en las juntas rotacionales (motores).



La calibración simplemente se logra realizando algunos lanzamientos y observando sus trayectorias. Así como también, moviendo el robot manipulador a algunas posiciones predefinidas.

Los autores realizaron demostraciones del sistema y los resultados fueron que 2 de cada 3 eventos fueron interceptados exitosamente por el robot. La mayoría de los fallos se debieron al limitado campo visual horizontal de la cámara debido a que los objetos

---

eran detectados demasiado cerca al efector final del robot.

Se concluye que es posible interceptar con un robot-manipulador pelotas en pleno vuelo lanzadas por seres humanos, utilizando visión por computadora y el filtro Kalman para predicción de la trayectoria de la pelota. Esto se logra cuando se diseñan cuidadosamente los algoritmos convenientes para implementarlos a altas velocidades. Se generaron predicciones muy rápidas porque también se seleccionó una posición buena de la cámara y se logró la calibración necesaria introduciendo a un modelo algunas trayectorias estudiadas de la pelota. ("Calibrating by Doing").

En [1.3] se presenta una nueva aproximación a la planeación de movimiento de robot-manipulador en línea, para la interceptación de objetos móviles. Esto es un esquema híbrido, el cual combina la técnica de Navigation con un método convencional de seguimiento de trayectorias. La implementación de la técnica propuesta es discutida a través de un ejemplo de simulación. Se experimentó con simulaciones realizadas en una computadora personal (PC). Por simplicidad se utilizó un robot planar de dos grados de libertad tipo SCARA. El objeto a interceptar se asume como una masa puntual moviéndose en plano X-Y. Las coordenadas X-Y del objeto se suponen disponibles para el sistema interceptor a través del sistema de visión. Así mismo, el módulo dinámico de SIMULINK<sup>®</sup> y MATLAB<sup>®</sup> fueron usados para las simulaciones. La tolerancia de agarre fue de 10 mm (1 % de la máxima distancia entre el robot y el objetivo), y la tolerancia de velocidad fue de 10 mm/s (2% de la máxima velocidad).

Este trabajo presentó un nuevo enfoque de la planeación de trayectorias de un robot manipulador en línea para interceptar objetos móviles. La técnica propuesta se basa en la técnica de navegación (navigation-based technique) que es robusta y computacionalmente eficiente para la interceptación de objetos que maniobran rápidamente. La técnica de navegación utilizada es una mejora a la técnica llamada Ideal Proportional Navigation Guidance (IPNG). Esta técnica ha sido modificada para robots interceptores para reflejar algunas capacidades de maniobra de los robots-manipuladores sobre los misiles. Una diferencia entre los dos es que la velocidad del robot y la del objetivo deben coincidir sin impacto entre ambos para atrapar el objeto sin dañarlo.

Para ello, se desarrolló un esquema de interceptación híbrido que combina la técnica de interceptación navigation-based con un método convencional de seguimiento de trayectorias para atrapar objetos que maniobran rápidamente. La técnica propuesta se discute a través de la simulación de un ejemplo por computadora. Ilustrándose que el método híbrido arroja resultados favorables sobre los métodos convencionales de seguimiento de trayectorias de objetos móviles puros. Por otra parte, Robot Hand/Eye Coordination y Active Vision son dos áreas que llaman la atención [1.4]. En este caso, se ha examinado el uso de visión para manipular directamente al robot. Específicamente, investigaciones en el MIT han combinado visión y manipulación robótica aplicada al seguimiento y captura de pelotas lanzadas en ambientes controlados (conocimiento de los factores ambientales: velocidad del aire, temperatura, humedad, gravedad, etc). Este trabajo incorpora un nuevo sistema de visión, el cual requiere de un mínimo de ambiente controlado e implementa nuevos métodos para el seguimiento de objetos, calibración de robot/cámara y nuevos algoritmos de captura

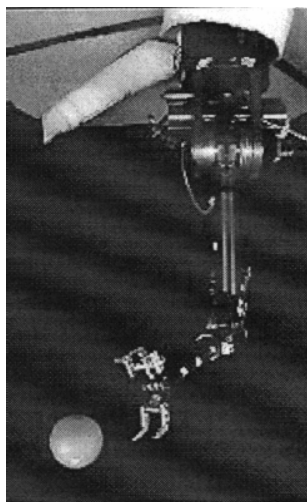
(catching algorithms). Es importante mencionar que en un ambiente controlado se conocen o se estiman variables del medio ambiente, como son; velocidad del aire, humedad, iluminación para el sistema de visión, etc.

El robot-manipulador utilizado es de 7gdl controlado por cable y un sistema de visión activa (se trabaja solo sobre el área de interés de la imagen) anclado en el techo. Contiene dos cámaras a color CCD (Charge Coupled Device) montadas cada una sobre un sistema actuador mecánico de dos grados de libertad. El procesamiento de la visión es hecho utilizando simple detección por gota para localizar objetos de color. El objetivo de esta investigación es desarrollar los métodos de control y algoritmos adicionales requeridos para completar la captura de objetos ligeramente lanzados.

Los elementos necesarios son:

- 1) Un sistema de visión capaz de localizar objetos.
- 2) Controladores y métodos de rastreo robusto implementados al sistema de visión activa para rastreo rápido del movimiento de objetos.
- 3) Calibración entre el sistema de visión y robot-manipulador.
- 4) Predicción de la trayectoria de los objetos lanzados, y
- 5) Generación de la trayectoria del robot-manipulador para la intercepción del objeto.

Este trabajo presenta también técnicas de localización y rastreo robusto de objetos usando información visual, el algoritmo recursivo de mínimos cuadrados para predicción de la trayectoria basado en el modelo (A recursive least squares algorithm for model-based path prediction) del objeto lanzado. Además, se discuten métodos para la determinación de puntos seguros de intercepción y una nueva técnica polinomial de generación de trayectoria para el robot interceptor.



**Figura 1.3** Robot de 7 gdl y sistema de visión montado en el techo llamado Eye-Fast-Gimbal.

El autor concluye diciendo lo siguiente: El sistema fue probado con lanzamientos del objeto esférico a mano desde diferentes orientaciones a una distancia aproximada de

1.5 a 2.5 metros hasta el brazo del robot. El tiempo promedio del viaje fue de 0.5 s. Los mejores resultados de desempeño fueron del 70-80% de lanzamientos similares, y en particular,

- 1) Se presentaron la implementación de varios controladores para el sistema de visión activa, incluyendo el controlador adaptivo por modelo de referencia (Model Reference Adaptive Controller), los resultados del desempeño de los controladores PD y el adaptativo, muestran una mejora en el controlador adaptativo para el seguimiento de trayectorias.
- 2) Se presentó un método para determinar la localización del objeto.
- 3) También se propuso un método para visualizar el seguimiento (tracking) de objetos móviles, el cual es capaz de seguir el objeto a una velocidad de más de 4 m/s a una distancia de 1 m desde las cámaras (8 m/s a 2 m, etc.).
- 4) Un método simplificado de calibración que coordina el manipulador y al sistema de visión. Los resultados de calibración tienen una exactitud de aproximación de 5 mm sobre todo en el espacio de trabajo del robot manipulador.

Utilizando el nuevo sistema de visión activa y de calibración, se obtuvieron resultados satisfactorios para el robot interceptor. Los elementos involucrados en la intercepción fueron:

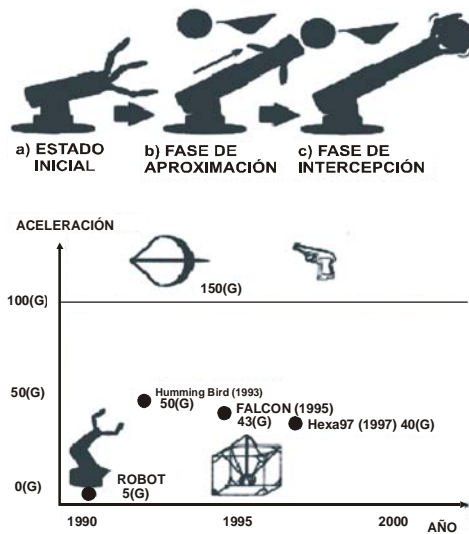
- 1) La implementación de la técnica fijando los datos de la pelota lanzada en una trayectoria parabólica para la predicción del lanzamiento.
- 2) Se presentó un método estructurado para predecir el camino del lanzamiento que determina los puntos seguros de intercepción de la trayectoria.
- 3) El reemplazo de un filtro de segundo orden para generación de la trayectoria por un filtro polinomial de tercer orden para generar mejores trayectorias de seguimiento del efector final del robot manipulador.
- 4) Un cronometrado preciso para el cierre del efector de sujeción (gripper) y métodos de orientación para el efector final fueron implementados para mejorar los resultados de intercepción.

Utilizando todas las técnicas mencionadas arriba, el sistema logró resultados satisfactorios al interceptar (catching) esferas en pleno vuelo libre. El porcentaje de intercepción, en el mejor de los casos, fue de aproximadamente 70-80%. Y se encontró que el controlador adaptativo, el cual mostró el menor error de seguimiento, fue de vital importancia para los algoritmos de intercepción.

El diseño de un sistema de captura con una respuesta extremadamente grande se analiza en [1.5]. La clave para alcanzar esta respuesta está en el mecanismo de acoplamiento de brazo y del efector de sujeción del robot (AGCM), donde la energía potencial inicial se acumula en el brazo y es convertida a energía cinética del brazo y contribuyendo con energía cinética para cerrar el efector final en el punto de captura sin ningún tiempo de retardo. Los resultados experimentales muestran que la máxima aceleración es de 91 G (veces la gravedad de la tierra), y el tiempo total de captura de



0.03s. El experimento más complejo fue capturando pelotas en caída libre.



**Figura 1.4** Comparación de aceleración en robots convencionales.

Los autores concluyeron que se analizó el diseño de un sistema de captura de objetos a alta velocidad utilizando como recurso la transformación de energía. Se propuso el mecanismo de acoplamiento brazo-gripper (AGCM), de tal forma que la energía inicial acumulada se transmite del brazo al gripper. Los resultados experimentales del desarrollo del robot muestran la máxima aceleración alcanzada 91G, rompiendo por casi dos veces la marca mundial, y el tiempo total de captura de 0.03, experimentando con la captura de una pelota. Este proyecto fue apoyado por la concesión Creating a Brain Project of Core Research Evolutional Science and Technology (CREST), JST, Japan.

Los algoritmos basados en el comportamiento humano para cachar pelotas en pleno vuelo han sido investigados y estudiados en [1.6]. En este trabajo, la validación de dichos modelos son probados “cachando” pelotas que son arrojadas desde arriba verticalmente. El modelo OAC (Optical Acceleration Cancellation) y el modelo OAC invertido son simulados primero, y después probados experimentalmente con un robot móvil. Los resultados numéricos y experimentales muestran que ambos métodos son capaces de interceptar una pelota lanzada, pero los movimientos inicial y final son diferentes. Un nuevo programa de procesamiento digital de imágenes. DVRobot fue escrito para el procesamiento rápido de imágenes digitales.

Los autores concluyen lo siguiente: En las simulaciones y experimentos, manteniendo la pelota en subida a proporción constante o cayendo a una aceleración constante en el plano de la imagen (positivo y negativo  $C_1$ ) se determinan estrategias para cachar un objeto en caída vertical. La experimentación valida las simulaciones mostrando que el movimiento inicial del robot es rápido, cuando  $C_1$  es positivo, en contraste cuando  $C_1$  es negativo la velocidad del robot drásticamente disminuye. Los resultados de esta investigación aporta entre la psicología perceptual y la robótica, el desarrollo y evaluación de algoritmos perceptuales humanos que permitirá explorar y cuantificar

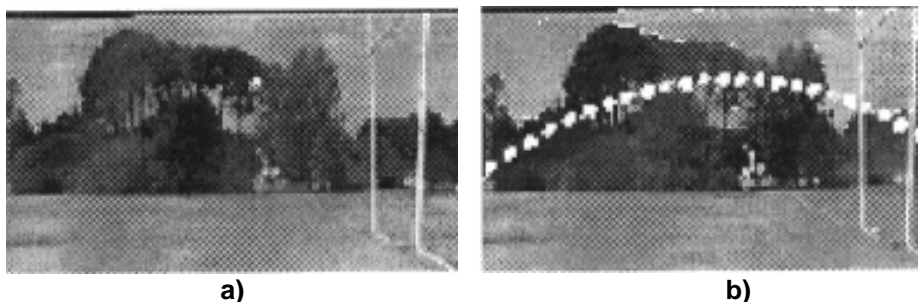
sistemáticamente los límites de los parámetros de control variable en el plano de imagen, para las condiciones de un mundo real.

Los algoritmos de visión basados en visión humana para interceptar pelotas u objetos en pleno vuelo en una dimensión han sido investigados y estudiados [1.7]. Se desarrollaron nuevas estrategias perceptuales para el espacio tridimensional que son simples y robustas. Los resultados de las simulaciones han mostrado la viabilidad de los algoritmos de visión humana para interceptar. Este trabajo trata la validación experimental de dos algoritmos perceptuales: Linear-Optical-Trajectory (LOT) y la Optical Acceleration Cancellation (OAC). Los autores concluyen lo siguiente: Se investigaron los principios de percepción tales como: constant image rate (OAC), y angular constancy en el plano de la imagen (LOT), aplicando estos principios a la persecución de navegación. Con estos principios, los algoritmos de control por retroalimentación visual (visual-servoing) permitirán a los robots móviles interceptar proyectiles con trayectorias complicadas y variables. Además, se desarrollaron algoritmos utilizando ambos modelos OAC y LOT propuestos por psicólogos en tres dimensiones.

El algoritmo OAC implementado sobre el robot es capaz de mantener el error lateral entre la pelota y el robot cercano a cero. Y puede moverse hacia atrás y hacia delante para cancelar la aceleración óptica. La estrategia LOT mantiene la posición de la pelota directamente en el plano de la imagen usando sólo información geométrica, y también puede disminuir el error lateral entre la pelota y el robot a valores cercanos cero.

En la simulación con una trayectoria parabólica perfecta, la imagen de la pelota no sólo se eleva a razón constante del plano de la imagen también la posición de la pelota sobre el plano de la imagen en movimiento sigue una línea recta. Se considera que una investigación multidisciplinaria entre la psicología perceptual y la robótica nos ayudará a desarrollar algoritmos de navegación que serán más robustos, poderosos y simples.

El problema relacionado con el rastreo (tracking) a una pelota en pleno vuelo a través de un ambiente incontrolado se analizó en [1.8]. Mientras trabajos anteriores que tratan el rastreo de pelotas en ambientes estructurados han sido satisfactorios, la complejidad aumenta cuando se consideran casos más generales. Un modelo de conducta generalizado es desarrollado para predicción, utilizando el Filtro discreto de Kalman, para refinar los parámetros de vuelo de acuerdo con la información ganada de cada imagen. La extrapolación de la trayectoria es lograda por predicción iterativa utilizando la última posición y velocidad conocidas.



**Figura 1.5** Vuelo de la pelota en un ambiente incontrolado.

Se presentaron y se discutieron los resultados del análisis de una muestra de secuencias de imágenes, y se sugirieron restricciones para mejorar la exactitud como mejoras en un trabajo futuro.

Los autores concluyen lo siguiente: La investigación exploró un área del rastreo de una pelota moviéndose libremente en un ambiente incontrolado utilizando el filtro de Kalman generalizado. En aplicaciones previas en ambientes controlados da una buena segmentación (procesamiento de imágenes), y cuenta con el conocimiento de una situación específica para lograr un rastreo exacto de una pelota en vuelo.

La flexibilidad se incrementó considerando escenas del exterior y diferentes velocidades y tamaños de pelotas. Sin embargo, la exactitud puede disminuir como resultado de una segmentación inconsistente de la imagen debida a los efectos de la luz y la interferencia del fondo. Además, factores como el viento y el giro de la pelota pueden afectar su vuelo en la realidad y se debe considerar si fue lograda una predicción adecuada o no. El conocimiento previo de estos factores se incorporó en el modelo de rastreo (tracking) satisfaciendo la situación particular. Pero esto reduciría la habilidad de rastreo en casos generales. Para mantener flexibilidad y permitir un ajuste automático para cada nueva situación, se deben detectar anomalías en el vuelo de la pelota en cada imagen para actualizar los parámetros del modelo de predicción.

El sistema desarrollado suministra las bases para un sistema de rastreo general, permitiendo flexibilidad en el ángulo de la cámara, el ambiente y las velocidades y tamaños de las pelotas, y es capaz de rastrear una pelota dentro de su área de aproximación en la mayoría de los casos, dando una indicación de la confianza de predicción. Sin embargo, el incremento de la exactitud puede ganarse utilizando técnicas de segmentación mejoradas, e incorporando los atributos de las pelotas por el arrastre del aire.

En [1.9] se presenta un sistema integrado, el cuál es capaz de perseguir un objeto móvil y mantenerlo centrado en la imagen, esto se logra controlando el efector final de un robot. El sistema está integrado por tres partes; detección de movimiento independiente, rastreo, y control del efector final del robot. El trabajo se concentró en el mecanismo de detección y discute brevemente los problemas de control y rastreo. El sistema corre continuamente en el tiempo y actualiza la localización del objeto a una velocidad de 25 Hz. El objeto móvil puede rastrearse, aunque el observador realice movimientos independientes y desconocidos, combinando los dos movimientos de traslación y rotación.

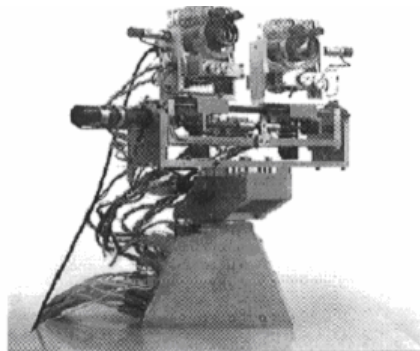
Se trabaja con un algoritmo simple de detección de movimiento, debido a que el costo computacional es de mayor importancia para un sistema de tiempo real con retroalimentación. El algoritmo es interactivo y económico computacionalmente. La complejidad en el tiempo de ejecución es  $O(n)$  para una imagen que contiene  $n$  píxeles. El procesamiento de la imagen se realiza en el procesador de tubería MaxVideo200 y el algoritmo de control del efector final corre con una computadora Transputer T800 network. Se presentan algunos resultados experimentales fuera de línea, donde se realizaron comparaciones entre los modelos de movimiento de imágenes traslacionales y afinidad de velocidad.

Los resultados arrojados fueron los siguientes: Se estudió la diferencia del desempeño entre los modelos de afinidad de velocidad y traslacional en experimentos fuera de línea. En diversas situaciones, el modelo de traslación es suficiente, debido a que la velocidad de cuadro es satisfactoriamente alta, y el movimiento del objeto rastreado es alto comparado con el movimiento causado por la expansión de la imagen.

Además, se presentó una aplicación en tiempo real de un sistema integrado en una plataforma head-eye. Este sistema utiliza el modelo traslacional y es capaz de detectar y rastrear un objeto en movimiento en una escena donde la cámara está en movimiento o estacionada, a una velocidad de cuadro de 25Hz. El algoritmo es simple, computacionalmente económico y conveniente para aplicaciones en tiempo real. Conectando la aplicación de rastreo de objetos al sistema de control del Head-eye, se generó un sistema de persecución de objetivos de gran eficiencia con capacidad de mantener un objeto móvil centrado en la imagen. Esto quedó demostrado.

Durante los experimentos se encontró que las limitaciones en el modelo de velocidad traslacional, lo hacen menos conveniente para lentes de gran ángulo de visión, al menos para lograr un buen rendimiento cuando el movimiento de la plataforma causa una expansión de imagen. Para un sistema que opera continuamente en el tiempo, éste ha demostrado tener una mayor importancia para mantener una alta velocidad de marco (frame-rate). Este hecho es aún más enfático para un sistema de control por retroalimentación "head-eye" motorizado desde el algoritmo de procesamiento de imágenes.

En [1.10], se utilizó el filtro de Kalman para atenuar las oscilaciones y lograr un movimiento suave del sistema seguidor. El sistema MDOF es un sistema de visión activa binocular con buena eficiencia y con un gran número de grados de libertad. Su funcionamiento depende principalmente de dos aspectos: del procesamiento de las imágenes y el control.



**Figura 1.6** El sistema binocular Multi-Degree of Freedom (MDOF).

En este trabajo se caracterizó el desempeño del sistema monocular para un seguimiento suave. Este sistema es utilizado para seguir binocularmente objetivos en un ambiente de vigilancia. Uno de los aspectos de esta caracterización fue la inclusión del procesamiento de visión. Para efecto de control, se utilizaron cuatro tipos de entrada estándar: step (impulso), ramp (rampa), parábola (parabólica) y sinusoid (senosoidal). Las respuestas obtenidas del sistema pueden usarse para identificar los subsistemas

que pueden optimizarse. Mostrando además, que la predicción y la estimación de la velocidad son esenciales para un buen desempeño en el seguimiento de objetos (tracking).

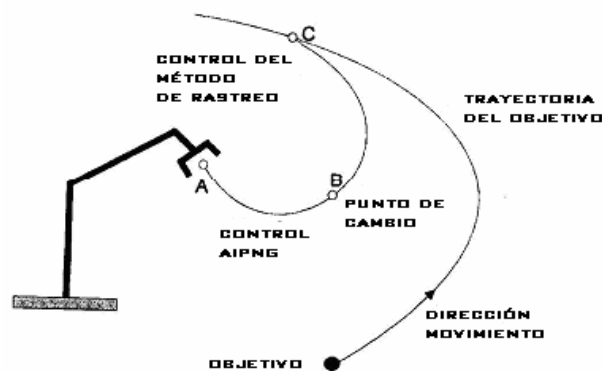
Los autores concluyeron lo siguiente: En este trabajo se presentó la caracterización del comportamiento visual del seguimiento suave de objetos utilizando señales típicas de entrada al sistema de control. La persecución suave está basada principalmente en el cálculo del flujo diferencial. Se consideró el tiempo de respuesta del sistema con las entradas antes mencionadas y se logró especificar las limitaciones de desempeño del sistema. No solamente se caracterizó el desempeño del control de la estructura servo mecánica, sino también el desempeño de las rutinas para el procesamiento de las imágenes. Los resultados permitieron la identificación de las no-linealidades específicas del sistema y mostramos que la predicción es esencial para el comportamiento del seguimiento.

Una nueva técnica para la intercepción de objetos en movimiento por medio de robot autónomo se presenta en [1.11]. En este caso, la técnica utilizada es Navigation-Guidance-based que es robusta y computacionalmente eficiente para la intercepción de objetos que maniobran rápidamente (que varían su movimiento en forma rápida y al azar). Ésta se utilizaba para controlar los misiles que interceptaban objetivos móviles. El problema que presenta el robot es que debe interceptar el objeto igualando suavemente la velocidad de éste (smooth grasp). La propuesta es desarrollar una hibridación de dos técnicas de intercepción: navigation-based interception con la técnica conventional trajectory tracking method, con el fin de interceptar objetos que maniobran rápidamente. Consecuentemente, la posición y velocidad instantánea del objeto se obtiene por retroalimentación visual.

En la técnica Slow-maneuvering objects: predicción, planeación y ejecución (PPE), el robot se adelanta a la posición predicha. Funciona, pero no para objetos que maniobran rápidamente. En la técnica Fast-Maneuvering Object se han utilizado diferentes técnicas con retroalimentación visual. (visual-feedback-based tracking systems), pero el desempeño se ve afectado por las restricciones dinámicas del robot. Para compensar el tiempo de retardo computacional, el estado del objeto se predice poco antes de interceptado. Los métodos mencionados no pueden hacer que robot genere trayectorias mínimas para la intercepción de objetos que maniobran rápidamente. Otros métodos navigation and guidance se aplican a los misiles que interceptan aviones evasivos. Son técnicas para optimizar el tiempo de intercepción. Aquí el robot puede moverse en cualquier dirección, mientras los misiles sólo pueden acelerar paralelamente a la dirección de su velocidad. Existen dos técnicas que el autor considera PNG (Proportional Navigation Guide) que es utilizada para interceptar objetivos con misiles. Esta técnica ya conocida en la comunidad científica consiste en nulificar la velocidad angular de la línea de vista (LOS). La otra técnica se llama Ideal Proportional Navigation Guidance (IPNG) que es una mejora de la PNG. Estas técnicas son usadas para impactar el objetivo, y se emplearon por primera vez en 1991, no se investigó cuando se igualan las velocidades de intercepción para evitar el impacto como un problema. Posteriormente en 1999, se utilizó IPNG junto con el método convencional de rastreo y lo llamaron PD-type Computed-Torque control Method. Este garantizaba igualar la posición y velocidad en el punto de intercepción.

La técnica PNG genera resultados de tiempo óptimo cuando rastrea objetivos en pleno viaje. Por otra parte, IPNG funciona para objetos que maniobran rápidamente (cuando su aceleración es predecible), y se asume que el interceptor y el objetivo solo pueden acelerar lateralmente en la dirección de su velocidad y además la magnitud de aceleración del objetivo es constante. Es por esta razón, que se introduce AIPNG Augmented Ideal Proportional Navigation Guidance como una mejora a IPNG.

Para dar solución al problema de obtener el tiempo óptimo de intercepción de objetos que maniobran rápidamente en la industria, se plantea un robot de seis gdl y una banda transportadora con velocidad desconocida y variaciones imprevistas. La posición y orientación de los objetos se realiza a través de visual-feedback y se considera que ya es recibida por el módulo de planeación de trayectorias del robot. Se ha modificado IPNG para que el robot no colisione con el objetivo. Entonces, se combinan las dos técnicas la IPNG que alcanza rápidamente el objetivo y la otra que permite al robot interceptar el objetivo con velocidad relativa cero. Es decir, sin impacto.



**Figura 1.7** Se muestra la técnica híbrida en donde el robot se mueve bajo el control de AIPNG y en un punto B, una técnica convencional de rastreo toma el control (la posición del efector final del robot con el objetivo es minimizado continuamente).

Desde antes de los 90's, se han realizado experimentos tecnológicos con robots espaciales ROTEX que han volado con el transbordador espacial COLUMBIA (STS55) [1.12]. En términos generales esto es un robot equipado con multisensores que está instalado en la plataforma de la nave, y trabaja exitosamente en modo autónomo, siendo teleoperado por astronautas, así como también, desde la tierra.

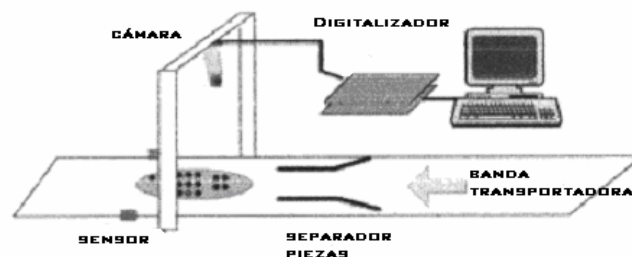
Estos métodos también incluyen teleoperación en línea y técnicas de programación para los sensores, involucrando el aprendizaje enseñados en un ambiente virtual. Las tecnologías importantes para lograr el éxito de los experimentos han sido: La tecnología de los multisensores en los efectores finales o grippers, conceptos de control por retroalimentación sensorial local y la simulación de compensación gráfica 3D de retardos (predictive simulation) desde la estación terrestre de telecontrol.

Estos conceptos no son más que ideas o propuestas, pero prueban su eficacia en un

vuelo espacial real, el saber cómo aplicarlos permite llevar a cabo las misiones de robots espaciales. Este trabajo se enfoca en la telerrobótica así como en los conceptos de simulación de gráficos en 3D, combinando imágenes e información de otros sensores para realizar la investigación de la propuesta de TeleSensorProgramming. El escenario de la misión de un prototipo reparando un satélite, incluye las estrategias para capturar satélites.

Se logró establecer que el concepto de TeleSensorProgramming, es una manera fácil de programar al robot fuera de línea en forma robusta, con estructuras de control basadas en sensores. La tarea propuesta mostró que el paradigma de la programación es también aplicable al campo de la telerrobótica, con largos tiempos de retardo, en donde la autonomía del robot es indispensable.

La investigación TeleSensorProgramming ha sido propuesta para ser utilizada en proyectos experimentales de servicio a satélites. El enfoque principal de este trabajo fue en el concepto del operador de una estación terrestre, incluyendo estrategias de control y supervisión para manejar los tiempos de retardo. Así como también, las incertidumbres en los ambientes de misiones prototipo. Muchas aplicaciones industriales requieren en sus sistemas automatizados procesamiento visual y clasificación de piezas apartes que se mueven sobre una banda transportadora. En [1.13] se aborda la selección de piezas basada en percepción para el procesamiento visual. La novedad es que en lugar de procesar toda la imagen, solo las áreas que son de interés y que requieren de atención, son analizadas.



**Figura 1.8** Clasificación visual instalado sobre una escena industrial.

La secuencia elegida puede ser usada por una gran variedad de tareas, incluso en la determinación de la forma de la pieza. Como sólo una porción de toda la imagen es procesada, el procesamiento visual puede darse en tiempo real y a la vez ser flexible sin requerir un equipo especial. Se describen además dos aplicaciones diferentes que se basan en esta investigación. En la detección de piezas defectuosas se explicará con detalle como se utiliza la secuencia elegida de la imagen. Como segunda aplicación, la investigación se ha llevado a cabo en un clasificador automático controlado remotamente en una planta manufacturera de televisores. Confirmando así la aplicación práctica.

Los autores concluyen que dentro del problema general de la clasificación automatizada de objetos o piezas sobre una banda transportadora, se propone un nuevo enfoque al procesamiento visual. Esto es, con base en la selección de percepción (selective perception). Aquí, en lugar de considerar toda la imagen proveniente de la cámara, sólo

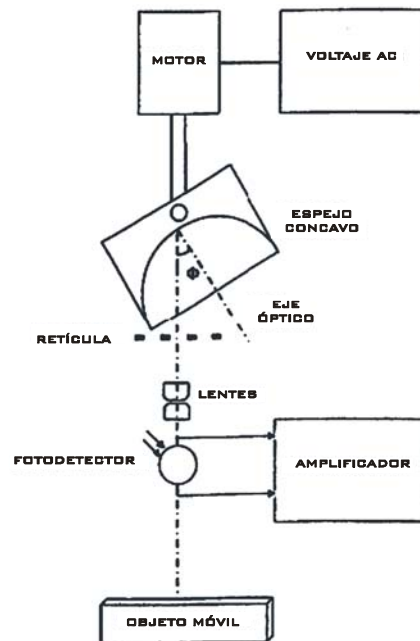
las regiones de interés son procesadas. Los recursos computacionales requeridos, por ser flexibles y trabajar en tiempo real simultáneamente, se reducen considerablemente. En este trabajo, el procesamiento visual consiste de un continuo de fases pre-atentas y atentas. La secuencia de porción de imágenes, así generada representa información visual espacio temporal. Este proceso se sigue de vez en cuando por la cognición, donde la secuencia de imágenes elegidas son procesadas de acuerdo a las demandas de la tarea en cuestión.

A continuación se describen dos aplicaciones basadas en esta investigación, ambas requieren de la toma de decisiones en tiempo real. En la tarea de detección de defectos de piezas, las secuencias elegidas son utilizadas para construir una representación de la pieza que sigue en la banda. La medición promedio es similar a ASSC entre la imagen previamente almacenada en la memoria de la máquina y la pieza a ser inspeccionada. Esto es evaluado para distinguir una pieza buena, de una con defectos.

Los valores de SAC (error promedio de posición) al borde del umbral son etiquetados como piezas defectuosas. En la segunda aplicación, se plantea una implementación de este estudio para la clasificación automatizada de piezas con controladores remotos en una planta manufacturera de televisores. Aquí, con un extensivo desarrollo y evaluación experimental se logró cerca del 100% de fiabilidad. Además, el procesamiento visual selectivo permite tiempos de inspección alrededor de 100 ms. Como un trabajo futuro, se plantea trabajar en la extensión de este marco para incorporar características más complejas para detectar otro tipo de defectos.

Por otra parte, en [1.14] se modeló un sistema de guía óptico y se estudia el procesamiento de señales y su operación. Este modelo se basa en el principio de considerar dos objetos en movimiento, el primero es el robot móvil y el segundo es el objeto buscado. El sistema descrito es capaz de detectar y medir el movimiento relativo del blanco, y responder generando señales eléctricas capaces de dirigir y guiar el vehículo sobre el piso. De esta forma, el robot respectivo o vehículo usado como dispositivo puede mantener el blanco del objetivo en la escena todo el tiempo. El sistema propuesto ofrece un alto grado de exactitud y fiabilidad desde que se considera el estado del arte en electrónica y componentes ópticos para el procesamiento de señales. Se utilizó el programa (ORCAD<sup>®</sup> 9) para simular el procesamiento de señales de un sistema de rastreo óptico y presentar resultados.





**Figura 1.9.** Arreglo óptico para el sistema de reporte de guía. La luz esparcida del objeto móvil es colectada por un espejo cóncavo y después se realiza filtrado especial, se enfoca en una fotocelda utilizando las ópticas apropiadas.

Se concluye lo siguiente: Se reporta un sistema automático de rastreo y guía óptico que puede mantener el blanco de un dispositivo relativo a un objetivo preseleccionado. Tales sistemas pueden ser útiles en varios campos. Por ejemplo, en sistemas para controlar automáticamente la sombra de las alas de un helicóptero sobre un objetivo específico. En otra aplicación, puede utilizarse para guiar un vehículo transportador cerca o hacia la posición de un objeto. La fase de modulación basada en la segunda y tercera armónicas promete proveer una señal de control excelente para la detección del movimiento relativo fijando el objetivo del objeto móvil

Por el método anterior, una excursión completa de la escena fuente puede ser detectada y las señales resultantes pueden disparar el circuito lógico. En el sistema existen dos demuladores, estos operan sobre cualquiera de las dos armónicas. Así como las segundas y terceras armónicas generadas por la fotocelda, para producir un voltaje por el filtro pasa bajas generando señales de para control. Los demuladores pueden usar cualquier grupo de armónicas y con propósitos ilustrativos; la segunda y la tercer armónica son adoptadas en este caso. Los resultados de simulación muestran que el desempeño del diseño reportado se realiza con precisión y puede ser empleado para una aplicación real.

---

## I.2 SISTEMA DE MANUFACTURA INTEGRADO POR COMPUTADORA (CIM).

### I.2.1 LAS MÁQUINAS AUTOMÁTICAS.

Las primeras máquinas automáticas datan del siglo XVIII, estas fueron desarrolladas para la industria textil. Hasta 1802 se dispuso de máquinas discretas, una de ellas desarrollada por M. Brunel para la fabricación de poleas [1.15]. El progreso de la automatización lo marcó la máquina *Transfer*, con base en un número determinado de estaciones de trabajo, cada una de las cuales realiza una tarea específica, montadas sobre una base común que dispone de un sistema de alimentación integral que traslada automáticamente el producto acabado de una estación a otra. La máquina integra dos partes esenciales en la automatización: la máquina automática y el transporte del producto entre máquinas.

Este principio de las *transfers* se utilizó en la industria del automóvil en 1924 [1.16]. Un importante paso en la automatización de procesos fue la introducción del movimiento continuo, Los sistemas continuos son aquellos en los que el flujo de materiales de los productos (agrupados si es necesario en familias homogéneas) recorre en forma ininterrumpida las instalaciones y procesos a ellos dedicados, desde las entradas hasta las salidas, como es usual. Por ejemplo; en las cadenas de montaje y en los procesos químicos continuos, que permite procesar un producto mientras se traslada, siendo la industria del papel la primera en utilizarlo de 1804 a 1833. En 1913, Henry Ford creó una cadena de montaje de magnetos. Su éxito en la reducción del personal y mejora del producto, incrementó en 1920 la demanda de productos de alto volumen de producción, volviéndose el movimiento continuo un elemento imprescindible [1.17].

Después de la segunda guerra mundial, la necesidad de crear nuevos productos provocaba que el diseño de sus componentes comenzara a ser problemático y muy costoso. Implicando un papeleo y una comunicación verbal difícil. La solución vino con la aparición de la computadora [1.18]. El diseño de nuevas piezas ha venido a ser un punto clave en las industrias de productos discretos (la del automóvil, electrodomésticos, computadores por citar algunas) al reducir drásticamente el tiempo de creación de una pieza así como su conexión entre las máquinas que la fabrican. Finalmente, el control de procesos mediante computadora ha permitido mejorar espectacularmente la calidad, el grado de seguridad y reducir al propio tiempo el coste energético y los costes de producción y mantenimiento [1.19].

La incorporación de los robots en la industria ha cambiado el enfoque de la organización de los procesos productivos y la inversión en maquinaria especializada. Ahora la industria no requiere orientarse a procesos que realizarán seres humanos, sino a dispositivos versátiles y reprogramables. Por otro lado, en numerosos casos, no se precisa comprar máquinas dedicadas a una tarea concreta, ya que pueden ser substituidas por un conjunto de robots. Este es el caso de las máquinas antiguas de soldar carrocerías de automóvil, que ahora se han convertido en una serie de robots que pueden ser reprogramados para soldar otros tipos de vehículos, sin que sea

necesario cambiar la máquina [1.20].

Según [1.17] dentro de las fábricas de productos discretos, se produjeron mayores cambios, teniendo incidencia en: el diseño del producto, la planificación de la producción, la producción de piezas, su manipulación, su inspección, el ensamblaje de partes de objetos y la organización de la información de la producción. En el diseño del producto, el Diseñador Asistido por Computadora (CAD) podrá generar piezas más adecuadas, según los criterios que se establezcan y las restricciones propias del proyecto.

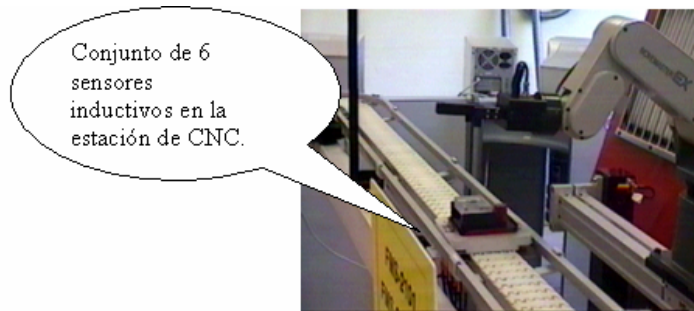
La planificación de la producción es un aspecto de gran importancia en la fabricación. En las fabricas automáticas de nuevo se utilizará la interacción con la computadora para lograr el plan óptimo de producción a partir de los datos de las piezas diseñadas, de los recursos que dispongan y las prioridades de cada trabajo. La producción de piezas se realizará en una o varias estaciones de trabajo controladas por computadora. En esta fase, los sistemas de control juegan un papel importante así como los de corrección de errores e imprevistos.

En el ensamblaje de piezas, máquinas especializadas, robots y otros dispositivos agrupados en estaciones de trabajo realizarán las tareas que de nuevo serán controladas por la computadora de proceso. En la figura 1.10, se muestran un PLC central que es el encargado de coordinar, enviar y recibir información de todos los módulos de un Sistema de Manufactura Integrado por Computadora(CIM), a través de líneas de comunicación modsoft. Las alarmas y los sensores envían señales a este PLC y éste a su vez responde dependiendo de la tarea que se esté llevando a cabo.



**Figura 1.10** PLC Central de un Sistema de Manufactura Integrado por Computadora (CIM).

La manipulación de piezas. Es decir, su carga, descarga y transporte se realizará por diversos dispositivos, entre ellos; cintas o bandas transportadoras y robots, en la figura 1.11, se puede observar que la estación de FMS (Sistema de Manufactura Flexible) tiene a un lado de la banda transportadora un conjunto de seis sensores inductivos que leen el número de vagón, esta información es enviada al PLC central, el cual se comunica con un programa que lleva el control y la información del vagón, como es: el número de vagón, es status, etc. Es decir, si contiene una pieza por maquinarse o si se trata de una pieza rechazada o aceptada, etc.



**Figura 1.11** Vagón en espera de ser atendido por el módulo de FMS. El robot mitsubishi® toma la pieza cuando esta se detiene en la estación.

La inspección de piezas se hará por medio de sistemas de percepción y dispositivos de medición, que conjuntamente controlarán la calidad de las mismas[1.22]. En la figura 1.12, se muestra el módulo de control de calidad o visión artificial. Este cuenta con una cámara, una lámpara especial de iluminación, un robot Mitsubishi® y una computadora PC, la cual contiene la tarjeta de adquisición de imágenes. Luego a través de un patrón de comparación se realiza el control de calidad dimensional de la pieza maquinada, una vez generado el reporte se envía al PLC central y se decide si la pieza es o no rechazada.



**Figura 1.12** Módulo de Visión Artificial.

Los almacenes automáticos alimentan al sistema o guardan los productos terminados, ya sean rechazados o aceptados. Generalmente son robot-manipuladores neumáticos los que alimentan al sistema. Finalmente está la automatización de la información de la producción que será gestionada por medio de programas de computadora, por ejemplo para controlar las existencias y los pedidos, y planificar las diversas tareas [1.23], la figura 1.13, muestra el módulo de almacén automático en él se utilizan políticas de almacenamiento, "lo primero que entra es lo primero que sale", o "lo primero que entra es lo último que sale". El robot neumático almacena producto en proceso de maquinado y/o producto terminado



**Figura 13.** Módulo de Almacén Automático.

## **I.2.2 SISTEMAS INTERMITENTES**

Los sistemas intermitentes son aquellos en los que las máquinas e instalaciones son lo suficientemente flexibles para manejar una amplia variedad de tamaños y formas, o bien que la naturaleza de la actividad implica un cambio importante y cíclico de materiales y productos, que se manejan por lotes a lo largo del tiempo, por lo que las piezas y productos se mueven intermitentemente como en los talleres organizados por tecnologías, en los que una orden que llega a una sección aguarda hasta que hay una máquina y un operario disponibles, entonces se ejecuta, aguarda su transferencia, pasa a otro centro de trabajo, etc.

Una solución propuesta, para lograr entre otros aspectos flexibilidad y modularidad de sistemas de manufactura a principios de los 90's, fue el paradigma CIM (Computer Integrated Manufacturing). Sin embargo, este paradigma fue desbordado por las expectativas, en una época carente de la tecnología requerida para su implementación, originando un alto costo en los sistemas informáticos requeridos para su integración [1.24].

Un nuevo paradigma para el control de la producción se conoce como MES por sus siglas en inglés (Manufacturing Execution Systems) en 1997; definido como un sistema computarizado integrado en línea, que consiste de una colección de herramientas y métodos para la gestión de producción. Este nuevo paradigma consiste en una serie de herramientas informáticas que requieren de una estrecha comunicación con el equipo físico de fabricación. En este sentido, el desarrollo de equipo en donde el robot permite la integración con el programa de control de manera modular y flexible fue de gran interés. En el presente trabajo, el término CIM se circunscribe al problema de integrar por computadora al equipo físico de fabricación con los métodos y herramientas de

producción. Además, en los sistemas CIM, la flexibilidad depende enormemente del desempeño de sus subsistemas de gestión, y su integración con la parte física del taller. Este último, es una de las partes más importantes del sistema consecuentemente, el diseño de los subsistemas de control **[1.25]**.

En la actualidad, los diferentes medios de producción requieren sistemas flexibles, no solamente en el ámbito de la generación de mejores métodos y máquinas más eficientes, si no también de una manera eficaz de integrarlos al sistema. Además, la mayoría de las empresas, tratan de implementar en sus sistemas de producción una rápida adaptación, por ejemplo; que una línea de producción de teléfonos celulares, sea capaz, con unos cambios mínimos al producto, añadirles alguna función extra, para mantenerse actualizados en el mercado consumidor **[1.26]**. El concepto CIM (Computer Integrated Manufacturing) implica la integración de todos los componentes (Personal, máquinas, información, etc) de una organización manufacturera. Una de las metas principales de integración contempla a la información, esta tarea requiere de la existencia del significado de la especificación de los datos y de los modelos de datos ampliamente.

La interfase de una base de datos permite una fácil comunicación entre humanos en el proceso de creación del modelo, el sistema de información debe ser flexible. Por lo tanto, la integración de la planeación de un sistema de manufactura y la planeación del proceso de ejecución de los propósitos del sistema de manufactura completo es principalmente un problema de integración de la información. Además, un modelo de datos se establece usando formalismos los cuales suministran una definición exacta de la aplicación "miniworld". Esta última contiene la estructura de datos, la cual incluye los objetos de aplicación, los atributos del objeto e información adicional, considerando las restricciones, operaciones implícitas, etc **[1.27]**. La configuración de un sistema de manufactura es una decisión estratégica que muchas compañías deben considerar cada vez que adquieren un sistema o modifican el actual, además de seleccionar los recursos necesarios para satisfacer mejor sus necesidades; lo que requiere el mercado, los productos nuevos, número de máquinas, características, herramientas, el desempeño del sistema, etc. es difícil de evaluar. Se utilizan metodologías para lograrlo, una de ellas es la descomposición jerárquica (hierarchical decomposition) **[1.28]**.

La configuración de un sistema de producción es un problema que ha sido investigado en los últimos años. Una línea de investigación enfoca los modelos estocásticos a Sistemas de Manufactura Flexible (FMS), los motivos son: por ejemplo, una máquina que falla y qué causa un mal funcionamiento del sistema completo, el rompimiento de una herramienta de corte, estos problemas son tratados por medio del modelo estocástico. En **[1.29]** se clasifican los modelos analíticamente de acuerdo a los correspondientes problemas tratados. Las cinco clases principales identificadas son: capacidad de procesamiento, capacidad de almacenamiento, fácil Layout, cantidad de pallets y sistemas de manejo de material. Reconociendo que el problema frecuentemente estudiado es la optimización de la capacidad de procesamiento en términos del agrupamiento y localización de las máquinas **[1.30]**,

En la figura 1.15 se muestra el módulo de FMS que se encuentra típicamente integrado por una fresadora y un torno de control numérico, para maquinar piezas en forma de

cubo y de cilindro respectivamente, estas máquinas se programan con lenguajes de programación antes de que el proceso inicie. Los programas se realizan en forma manual o apoyándose en algún programa de CAM. El robot mostrado es el que alimenta las máquinas y coloca las piezas maquinadas en los vagones que van sobre la banda.



**Figura 1.14** Sistema de Manufactura Flexible (FMS).

### **I.2.3 APLICACIÓN DE UN CASO REAL. DESCRIPCIÓN DEL SISTEMA.**

El sistema de producción es un sistema de manufactura automático similar a los FMS's, en el cual las piezas son maquinadas por centros de maquinado CNC. Las máquinas son idénticas y una o varias bandas mueven las piezas y herramientas a lo largo del sistema. Las piezas se mueven entre las estaciones de carga y descarga y las máquinas, mientras que las herramientas se mueven entre los almacenes centrales y locales de almacenamiento de las máquinas CNC. La diferencia principal con el tradicional FMS's, con máquinas paralelas, es que las piezas no son trabajadas por una sola máquina, sino por varias máquinas acomodadas en línea.

Las máquinas y accesorios son más pequeños que los tradicionales, el cubo de trabajo es de 400 mm. Esto puede ahorrar costos en la inversión de la compañía. En [1.31,1.32] se hace una descripción detallada.

Solo una pieza es montada en un pallet. Por lo tanto, el número de pallets moviéndose en el sistema corresponde al número de piezas en el sistema. Denotemos el término "mini-line" como un conjunto de máquinas que realizan operaciones en las mismas piezas. Cada mini-line puede maquinar uno o varios tipos de piezas. El sistema puede descomponerse de manera lógica en varias de éstas y sus configuraciones pueden modificarse durante la operación del sistema reduciendo los tiempos de colocación y montaje. Por ejemplo, un sistema se divide en dos mini-line: el primero maquina el producto A mientras el segundo maquina el producto B. El transportador de piezas mueve una pieza del buffer de entrada a la primera máquina de la mini-line; entonces cada pieza, después de haber sido maquinada por todas las máquinas se envía finalmente al buffer de salida.

El comportamiento del sistema depende de las líneas de flujo que comparten recursos comunes, como los transportadores de piezas, transportadores de herramienta, herramienta y estaciones de carga y descarga. Entonces se debe evaluar el desbalance de operaciones en las mini-líneas para estimar el desempeño del sistema [1.33]. Un sistema moderno de manufactura debe tener la capacidad de satisfacer las demandas de nuevos productos al mercado consumidor con un tiempo de diseño seguro. También, debe ser capaz de acomodarse al rápido desarrollo de las tecnologías computacionales. Es decir, perseguir la estructura modular abierta, la cual permite la flexibilidad, la integrabilidad y la simultaneidad entre los diversos elementos de manufactura [1.34].

Los robots son usados en muchas aplicaciones dentro de la industria y están relacionados directamente con la eficiencia en la producción. Por lo tanto, se utilizan ambientes virtuales de simulación para controlar el robot. Así como también, configurar los sistemas de robots con el fin de aumentar la eficiencia en la producción [1.35]. La creciente competencia globalizada requiere de manufactureras que entreguen productos competitivos con la mejor calidad y al precio más bajo. Uno de los retos más importantes que encara la industria manufacturera es la substancial reducción del tiempo utilizado en el desarrollo del producto [1.36].



**Figura 1.15.** Sistema de Manufactura Integrada por Computadora. Laboratorio de Mecatrónica UPIITA-IPN.

En la figura 1.15 se muestra un Sistema de Manufactura Integrado por Computadora (CIM), que cuenta con un módulo de Manufactura Flexible(FMS).



---

### I.3 PLANTEAMIENTO DEL CASO DE ESTUDIO

Actualmente ya han sido estudiados los sistemas robóticos que pueden seguir visualmente un objeto e interceptarlo, el cual viaja con velocidad desconocida sobre una banda transportadora.

En la literatura se han definido una variedad importante de problemas relacionados con el fenómeno de intercepción de piezas. La implementación de dispositivos robóticos con capacidad sensorial inteligente han sido reconocidos como sistemas autónomos esenciales en los sistemas de producción. Actualmente, un sistema típico de producción está compuesto por un robot, una banda transportadora y un módulo de maquinado. El robot requiere como información la velocidad con que opera la banda transportadora para interceptar el objeto.

Los objetos son colocados sobre la banda con orientaciones definidas y a intervalos. En estos sistemas, las dificultades aparecen cuando la velocidad de la banda varía aun siendo pequeña la variación o que el objeto no es colocado con suficiente precisión en su orientación. Por lo tanto, un sistema flexible que permita que el robot tome piezas de trabajo orientadas al azar y con velocidad arbitrarias sería un excelente problema para resolver, según [1.37].

Varias propuestas han sido estudiadas en el intento de desarrollar un medio de conseguir seguimiento de objetos basado en visión. Por ejemplo; el seguimiento de piezas sobre bandas transportadoras usando un sistema de control por aprendizaje [1.37]. La mayor restricción de esta propuesta es la necesidad de realizar numerosos ensayos para actualizar "aprender" la estrategia de control, sin olvidar la complejidad computacional que implica la implementación en tiempo real.

Otros autores han propuesto soluciones alternas a problemas similares, pero no han intentado interceptar dos o más piezas en forma múltiple. Es decir, que el efector final del robot tome una pieza, y después en la misma secuencia de movimiento tome la segunda pieza con la misma trayectoria calculada. Esto repercutiría en la recuperación de los tiempos que utiliza el robot y el controlador para calcular y ejecutar de nuevo la trayectoria para la segunda pieza.

Cabe mencionar, que esta idea nació al observar a los trabajadores en las empresas maquiladoras. En las líneas de producción donde el número de productos que viajan en las bandas transportadoras son muchos, para acomodarlos y empaquetarlos los trabajadores toman de dos o más piezas con una sola mano para colocarlos en las cajas, el utilizar esta forma de tomar las piezas les cansa menos y empacan más cajas en menor tiempo.

Con lo anterior, se pretende reducir los tiempos de espera de los vagones en las estaciones de los módulos del (CIM) incrementando como consecuencia la productividad del sistema de producción. Por lo tanto, problema a resolver es desarrollar algoritmos de intercepción múltiple de objetos. Además, utilizando el enfoque mecatrónico se plantea resolver el problema del fenómeno espacio-temporal de interceptar trayectorias múltiples conocidas, generadas por dos o más piezas en

movimiento.

Es importante, que los algoritmos estén definidos en función del tiempo, que sea factible fijar los tiempos de intercepción para cada partícula en forma independiente.

A la hora de implementar los algoritmos en forma real las restricciones serán las siguientes:

- No modificar la configuración física interna del sistema.
- No eliminar infraestructura del sistema.

Es importante mencionar que en la realidad muy pocos procesos pueden ser modificados crucialmente. Añadir o eliminar infraestructura es un proceso que resulta largo, costoso y sobre todo riesgoso. Por lo tanto, al utilizar el enfoque mecatrónico solo se modificará la programación del robot y se añadirá un sistema para retroalimentar la posición y velocidad de las piezas que viajan sobre la banda transportadora.

Es importante mencionar, que la simulación de sistemas de manufactura integrados por computadora (CIM), se ha vuelto una herramienta importante en el análisis dinámico de resultados de rendimiento de las máquinas y personal en una fabrica, incluyendo la determinación de cargas de trabajo en las líneas de producción, detección de la inactividad de los procesos, tiempos muertos, determinación de los cuellos de botella, así como de los recursos infrutilizados, flujo de material, etc. Estos sistemas de simulación permiten plantear estrategias de producción, plantear soluciones, reducción de costos. Por tal motivo, se plantea la implementación de estos algoritmos primero en simulación y se utilizará un simulador comercial para lograr tal fin. Ver figura 1.15.

#### **I.4 METODOLOGIA DE ESTUDIO**

A continuación se presenta la metodología para resolver el problema antes planteado:

- a) Diseño y desarrollo de los algoritmos de recuperación de tiempos (partícula seguidora), para interceptar al menos dos partículas libres.
- b) Implementación de los algoritmos a robot manipulador (Cinemática y dinámica Inversa).
- c) Diseño y desarrollo de simulador bidimensional para la intercepción de objetos vía robot interceptor.
- d) Simulación de la intercepción vía robot de dos partículas.
- e) Diseño y construcción de robot interceptor de 2 gdl.
- f) Implementación de la ley de control y pruebas de intercepción.
- g) Diseño y desarrollo de sistema de visión para obtener los parámetros cinemáticos de las partículas libres en sus trayectorias conocidas.
- h) Implementación y acoplamiento de las ecuaciones seguidoras en la cinemática Inversa del robot interceptor en la integración total del sistema.

Por lo tanto, la estructura en etapas de este trabajo de investigación es la siguiente: la primera etapa contiene los antecedentes de la intercepción de objetos utilizando robot

---

manipuladores, desde sus inicios (estado del arte) hasta sus aplicaciones, como son: en los deportes y en los sistemas de manufactura integrados por computadora.

En la segunda etapa se presenta el desarrollo de los algoritmos de recuperación de tiempos. Siendo el fenómeno del tiro parabólico el punto de partida como un caso particular. Además, se plantea la metodología de estudio y se validan los resultados con simulaciones utilizando un programa de cálculo formal y otro de simulación dinámica.

En la tercer etapa se integra al robot manipulador (prototipo) los modelos cinemáticos para la generación de trayectorias en la intercepción de los objetos en pleno movimiento bidimensional.

En la cuarta etapa, que abarca la parte experimental, se integra al robot manipulador un sistema para obtener los parámetros cinemáticos de las partículas a capturar, a través de una computadora personal (PC).

Finalmente, se probaran los algoritmos de recuperación de tiempos en un Sistema de Manufactura Flexible(FMS).

## **I.5 CONTRIBUCIONES PRINCIPALES DE LA INVESTIGACIÓN**

Las contribuciones principales de este trabajo de investigación son las siguientes:

*En la parte teórica:*

- 1.- Propuesta de una metodología para generar algoritmos de intercepción múltiple de trayectorias conocidas en un ambiente controlado.
- 2.- Desarrollo de los algoritmos de intercepción como un caso particular para el fenómeno de tiro parabólico con efecto de la gravedad, con posibilidad de controlar la posición del robot interceptor y la posición de disparo de las partículas en forma independiente.
- 3.- Implementación de los algoritmos de recuperación de tiempos a la cinemática inversa de un robot interceptor de dos grados de libertad utilizando el álgebra de Cuaterniones.
- 4.- Definición de las restricciones de las trayectorias de las partículas a interceptar en función del área de trabajo del robot interceptor.
- 5.- Aplicación de un esquema de control par calculado con PID para el control de movimiento del robot interceptor.
- 6.- Desarrollo computacional de un simulador de intercepción múltiple de trayectorias.

*En la parte experimental:*

7.- Diseño y construcción de prototipo de robot interceptor para validar la parte teórica, junto con la construcción de un disparador de dos partículas (objetos esféricos), a ángulos ajustables y velocidades de caída libre para validar la parte teórica.

## **I.6 SUMARIO**

En este capítulo se presentó el estado del arte de robot manipuladores que interceptan objetos en pleno movimiento. Lo anterior permitió plantear y definir el problema a resolver en este trabajo de investigación. Se describió en forma breve la historia de las máquinas automáticas hasta la concepción de los Sistemas de Manufactura Integrada por Computadora(CIM), en el cual se encuentran los Sistemas de Manufactura Flexible(FMS).

Se establece que la mayoría de los trabajos de investigación están orientados a minimizar los tiempos muertos en los sistemas de producción, esencialmente con la idea fija de incrementar la producción de las empresas con el fin de volverse competitivas y mantenerse en el mercado actual. Es decir, todas las innovaciones en los sistemas de producción van encaminados a incrementar la productividad reduciendo los tiempos muertos y los desvíos de recursos. La industria manufacturera se encuentra en una constante competitividad a nivel mundial debido a la globalización. Ahora, los productos sofisticados y versátiles son demandados por la sociedad consumidora, exigiendo a las industrias reducir al máximo los tiempos de diseño y producción ofreciendo mejor calidad y precios más bajos. En resumen, uno de los retos más importantes que encara la industria manufacturera es la sustancial reducción del tiempo utilizado en el desarrollo de sus productos.

Lo anterior se convierte en el punto principal de motivación para la realización de este trabajo. Además, se plantea el problema a resolver, la metodología de estudio y las aportaciones teóricas y experimentales de esta tesis doctoral.

## I. 7 REFERENCIAS

- [1.1] D, Hujic, E. Croft, G, Zak, RG. Fenton, J,K, Millis, B. Benhabib, "The robotic interception of moving objects in industrial settings: strategy development and experiment", Computer Integrated Manufacturing Laboratory, Department of Mechanical Engineering, University of Toronto, pags:1-29, January 1995.
- [1.2] U, Frese B. Bäuml. S. Haidacher. G. Schreiber, I. Schaefer. M, Hähnle. G. Hirzinger, "Off-the-Shelf Vision for a robotic Ball Catcher", Institute of Robotics and Mechatronics, German Aerospace Center, pags: 1-7, 2000.
- [1.3] Mehran Mehrandezh, Naftli M. Sela, Robert G. Fenton y Beno Benhabib, "Robotic Interception of Moving Objects Using an Augmented Ideal Proportional Navigation Guidance Technique", IEEE Transactions on Systems, Man, and Cybernetics –Part A: Systems and Humans, vol. 30. no.3, pags: 238-250, May 2000.
- [1.4] Won Hong, "Robotics Catching and Manipulation Using Active Vision", M.Sc. thesis, Dept. Mech. Eng., Mass. Inst. Technol., Cambridge, Sept. 1995.
- [1.5] Makoto Kaneko, Reika Takenaka y Masatoshi Ishikawa, "The Capturing Robot with Super High Acceleration".
- [1.6] Keshav Mundhra, Anthony Suluh, Thomas Sugar, Michael McBeath, "Interception a Falling Object: Digital Video Robot".
- [1.7] Anthony Suluh, Keshav Mundhra, Thomas Sugar, Michael McBeath, "Spatial Interception for mobile Robots".
- [1.8] Robin Hosie, Geoff West, "Predicting ball trayectories in uncontrolled environments" ,
- [1.9] Peter Nordlund y Thomas Uhlin, "Closing the Loop: Detection and Pursuit of a Moving Object by a Moving Observar" ,Julio, 1995.
- [1.10] João P. A. Barreto, Paulo Peixoto, Jorge Batista, Hetder Araujo, "Control Performance Issues in a Binocular Active Vision System".
- [1.11] Mehran Mehrandezh et al, "Robotic Interception of Moving Objects Using an Augmented Ideal Proportional Navigation Guidance Technique", IEEE Transactions on systems, man, and cybernetics, -Part A: Systems and Humans, Vol. 30, No. 3, pags: 238-250, Mayo, 2000.
- [1.12] K. Landzettel. B. Brunner, G. Hirzinger, "The Telerobot Concepts for ESS".
- [1.13] H.Isil Bozma, Hülya Yalcin, "Visual processing and classification of items on a moving conveyor: a selective perception approach", Robotics and Computer Integrated Manufacturing 2002.
- [1.14] Hossein Golnabi, "Simulation of an optical sensing system for automatic object tracking", Robotics and Computer Integrated Manufacturing 2002.
- [1.15] Bazañez L., Automatización Industrial, CEAM Revista de Economía Industrial No. 171. Marzo-Abril 1982.
- [1.16] Bright J.R., The Development of Automation, in Technology in Western Civilization vol. II M. Kranzberg and C. Pursell (eds) Oxford University Press,1977.
- 1.17] Gini G., Gini M., Automazione Programmabile e Fabrica del Futuro, Automazione e Strumentazione, vol. XXIX, No. 4, Abril 1981.
- [1.18] Dorf R.C., Introduction to Computer and Computer Science, 2da Ed. Boy and Fraser, San Francisco, 1977.
- [1.19] José Mompin Poblet, Sistemas CAD/CAM/CAE Diseño y fabricación por computador, Publicaciones Marcombo, México-Barcelona,1988.

- 
- [1.20] Albus J.S. y Evans J.M., *Sistemas Robots*, Investigación y Ciencia No. 1, Oct. 1976.
- [1.21] Bufo, E.S. y Taubert W.H., *Production-Inventory Systems Planning and control*, Homewood, R.C. Irwing, 1972.
- [1.22] IEEE Transactions on pattern Analysis and Machine intelligence. "A Real time automated Visual Inspection System for Hot Steel Slabs", Vol. PAMI 5, No. 6, Noviembre 1983.
- [1.23] Hax, A.C. y D. Candea, "Production Planning, and Inventory Control", Englewood Cliffs, Prentice Hall, 1984.
- [1.24] Zahfa Idelmérfaa, Jacques Richard, "CIM systems modeling for control system reusability". *INT. J. COMPUTER INTEGRATED MANUFACTURING*, VOL.11 No. 3. pags:195-204, 1998.
- [1.25] Jeffrey S. Smith, Sanjay. B. Joshi, "Reusable software concepts applied to the development of FMS control software". *Int. j. Computer Integrated Manufacturing*, Vol.5 No.3, pags:182-196, 1992.
- [1.26] Milton Joel B, *et al*, "Herramientas de apoyo orientado a objetos para la programación e interacción con robots en sistemas integrados de fabricación", *ELECTRO 2001*, Instituto Tecnológico de Chihuahua, Chih.
- [1.27] Santiago Arnalte, Rosa M., "An Information System For Computer-Integrated Manufacturing System", *Robotics & Computer-Integrated Manufacturing*, Vol.13, No. 3, pags:217-228, 1997
- [1.28] Dallery Y, Frein Y, "An efficient method to determine optimal configuration of a flexible manufacturing. system", *ANN Oper,Res* 1988; 15:207-25.
- [1.29] Dallery Y, "On modeling flexible manufacturing system using closed queuing networks". *Large Scale Systems* 1986; 11:109-19.
- [1.30] Matta A, Tolio T. "High flexible and high productive manufacturing system architecture". *Porceedings of the 15<sup>th</sup> International Conference of Production Research*, 1999.
- [1.31] Matta A. Tolio T. Karaesmen F. Dallery Y, "A new system architecture compared with alternative production systems", *Int J Prod Res* 2000;38: 17.
- [1.32] Semararo Q, Tabusso F, Tolio T, "Loading of a highly productive and flexible manufacturing system". *Proceeding of the Tenth International Conference on Flexible Automation and Intelligent Manufacturing*, M.D USA, pags:910.;9.June 26-28, 2000.
- [1.33] Sperling W, Lulz P. "Enabling open control systems An introduction to the OSACA system plataform". *ESPRIT III Project*, FtSW GmbH, Stuttgart, 1995.
- [1.34] Freund E, Rossman J, "Projective virtual reality: Brinttging the gap between virtual reality and robotics", *IEEE Trans Robotics Autom* 1999; 15(3):411-21.
- [1.35] L. Li, N. Schemenauer, X. Peng, Y. Zeng, P. Gu, "A reverse engineering system for rapid manufacturing of complex objeds", *-Robotics & Computer integrated Manufacturing*. Vol. 18, pags:5367, 2002.
- [1.36] Ren C. Luo, Robert E. Mullen Jr., et al, "An adaptive robotic tracking system using optical flow", *IEEE*, 1988.
- [1.37] W. T. Miller, "Sensor-Based Control of Robotic Manipulator Using a General Lerning Algorithm", *IEEE Jornal of Robotics and Automation*, Vol. RA-3, No.2, Apri1987.

---

## CAPÍTULO II

### MARCO TEÓRICO

#### II.1 GENERALIDADES

En este capítulo se presenta el estudio y desarrollo de los algoritmos de recuperación de tiempos llamados aquí funciones seguidoras para la intercepción de trayectorias generadas por partículas con movimiento libre, en condiciones ideales. Es decir, sin considerar fuerzas externas, excepto la causada por la gravedad. En el caso particular se estudia el fenómeno del tiro parabólico y en el caso general se plantea la metodología de estudio en forma de algoritmos para cualquier tipo de trayectoria.

Se presenta además, de la metodología de estudio, el análisis de posición, velocidad y aceleración de la partícula seguidora. Así como también, el análisis cinemático de la partícula libre. Para validar los resultados se presentan las simulaciones computacionales de cada caso y sus respectivas ecuaciones derivadas de las funciones seguidoras.

Con un ejemplo numérico se presenta el caso de intercepción múltiple de trayectorias de dos partículas bidimensionales, la metodología empleada y el desarrollo detallado de su ecuación seguidora.

Los resultados se presentan en forma de algoritmos computacionales que pueden implementarse en los modelos de la cinemática inversa de manipuladores robóticos comerciales para la intercepción de piezas que viajan sobre las bandas de los sistemas de manufactura integrados por computadora (CIM), utilizando retroalimentación visual para determinar la posición y velocidad instantáneas de las piezas montadas sobre los vagones.

## II.2 ALGORITMOS DE INTERCEPCION MÚLTIPLE DE TRAYECTORIAS BIDIMENSIONALES

### ANTECEDENTES

Desde inicios de la segunda guerra mundial el estudio del fenómeno de intercepción de trayectorias tomó auge en la milicia. En las guerras actuales fue evidente la capacidad de destrucción de objetivos múltiples con un solo cálculo. No obstante, las aplicaciones son versátiles. Por ejemplo, la implementación de las técnicas para el control de misiles seguidores de objetivos móviles, han sido aplicadas a los robot manipuladores para interceptar objetos. Aún más, en los sistemas de manufactura integrados por computadora, para realizar la tarea de palletización que es la alimentación de los módulos [2.1]. Una de las primeras aplicaciones fue en los juegos deportivos en los años 80's, con la creación de un robot manipulador que jugaba ping pong utilizando un sistema de visión que consistía de cuatro cámaras como ojos [2.2].

Además, casi al final de la segunda guerra mundial nacieron los requerimientos de sistemas guiados en forma táctica de misiles como resultado de los ataques *Kamikaze* contra E.U. Después de la guerra fue claro que las armas navales usadas sin sistemas de guía no fueron adecuadas para derribar a los aviones suicidas contra los barcos estadounidenses. En 1944, la armada de E.U. empezó a desarrollar los programas de guiado a distancia de misiles. Después de seis años de arduo trabajo, interceptaron el primer objetivo, un avión sin piloto [2.14].

Estos misiles usaron los conceptos de guiado que funcionan bien, no sólo contra objetivos estacionarios o predecibles, sino que también, eran efectivos contra aviones que ejecutaban maniobras evasivas, cuya posición futura era incierta. El área táctica innovó la tecnología de guiado y se adecuó a los misiles que tenían ventaja en aceleración sobre el objetivo.

La estrategia balística de misiles es diferente al guiado táctico de misiles, por que ellos viajan grandes distancias y están diseñados para interceptar objetivos estacionarios cuya ubicación es conocida. La estrategia de guiado táctico (tactical homing missile) detecta el objetivo con un sistema localizador e intercepta al objetivo. La intercepción estratégica no requiere de un sistema localizador ya que utiliza una fracción del vuelo total y utiliza la estrategia balística de misiles.

Los científicos alemanes, después de 14 años de trabajo en el año de 1944, implementaron sensores en el sistema de guiado con alcance de varios cientos de kilómetros. Sin embargo, los E.U. fueron los primeros en desarrollar equipo y programas en sus misiles, innovando la tecnología de misiles estratégicos [2.15]. Hoy en día la estrategia balística de misiles es exacta y viajan en rangos intercontinentales a altas velocidades.



### II.3 INTERCEPCIÓN DE OBJETIVOS MÚLTIPLES

La idea de la intercepción múltiple nació también en los últimos años de la segunda guerra mundial. Según [2.16] ambas técnicas: la táctica y la estratégica deben contar con la habilidad de alternar la localización del objetivo, ya que de no ser así puede ocasionarse un error de distancia (miss distance).

Por ejemplo, para interceptores tácticos, el interceptor se guía por el centroide de dos aviones volando en formación. Cuando un avión se sale del campo de visión del sistema de localización del misil, el otro avión puede ser interceptado. En este caso le parece al misil interceptor que el objetivo ha cambiado su posición instantáneamente, de la localización del centroide a la localización del avión interceptado. En otras palabras, hubo un cambio aparente en la posición del objetivo, que era el centroide y que ahora es el avión interceptado.

El sistema de localización puede entonces ver dos o más objetivos y asegurar de alguna manera la intercepción, aunque el costo sea una gran aceleración para cambiar su dirección en un tiempo pequeño.

En [2.17] el programa de planeación y búsqueda para guarda costas en los años 1970's, surgió de la necesidad de crear un programa para localizar barcos en problemas, cuyas señales de auxilio eran débiles. Para este efecto, el guarda costa con la poca información obtenida debía empezar la búsqueda y por lo tanto, estimar un plan estratégico. El sistema de planeación se construye a partir de un proceso estocástico el posible movimiento del objetivo desde la última posición conocida.

Produciendo una distribución probabilística de la localización del objetivo en función del tiempo. Estas técnicas fueron modificadas, extendidas y adecuadas para adaptar planes de búsqueda, y los procesos estocásticos se desarrollaron para predecir la posición y el movimiento de los objetivos en el tiempo.

De inicio se utilizaron filtros de Kalman pero esta técnica no pareció apropiada para este tipo de problemas; debido a que requiere aproximaciones y no acepta información negativa. Además, el filtro de Kalman produce distribuciones normales para la distribución del estado del objetivo. Esto es insatisfactorio por que se necesita calcular distribuciones probabilísticas de tipo arbitrario en vez de representar exactamente la localización del objetivo y planear una búsqueda efectiva.

Anteriormente, se empleaban métodos numéricos discretos para desarrollar el seguimiento de un objetivo, como la técnica no lineal Bayesiana. Ésta es llamada aproximación probabilística de seguimiento y fue desarrollada en 1970. El seguimiento de objetivos múltiples se llama problema de inferencia Bayesiano. Esta técnica para detección y seguimiento no es muy conocida en la comunidad que tratan el problema de seguimiento de objetivos múltiples.

De lo anterior, se propone en este trabajo una metodología para la intercepción múltiple de trayectorias vía robot interceptor, el cual ofrece ventajas, por que un robot puede moverse con más habilidad que un misil y con aceleración en cualquier dirección [2.3].

El fenómeno de intercepción se define como: "El acercamiento a un objeto móvil mientras se iguala su posición y velocidad en el tiempo más corto posible" [2.3].

Existen dos técnicas de intercepción vía robot interceptor, una de ellas es simplemente calcular el punto de intercepción, moverse y llegar antes que el objeto, esperar y atrapar en el momento apropiado (snatching). Los humanos usan esta técnica de la siguiente manera: utilizan al sentido de la visión y el sentido del tacto para determinar cuando cerrar la mano y atrapar el objeto móvil.

La otra técnica consiste en igualar la trayectoria y velocidad del objeto para atraparlo sin impacto y adquiriendo mayor tiempo para tomarlo en pleno vuelo (catching). Los humanos por lo regular no utilizan esta técnica. Sólo cuando se trata de objetos pesados y rápidos emplean una combinación de las dos técnicas anteriores [2.4].

La metodología de estudio que se propondrá, considera la técnica de intercepción con impacto (snatching) y sin impacto (catching) con una partícula y después con dos partículas. Cabe señalar, que estas técnicas son las más simples para interceptar un objeto, pero la contribución de este trabajo radica en que se atraparan dos objetos de manera múltiple y esta será la parte de interés.

Consecuentemente, se iniciará el estudio del fenómeno de intercepción con el análisis del fenómeno del tiro parabólico con efecto de la gravedad, aunque también puede aplicarse a cualquier tipo de trayectoria.

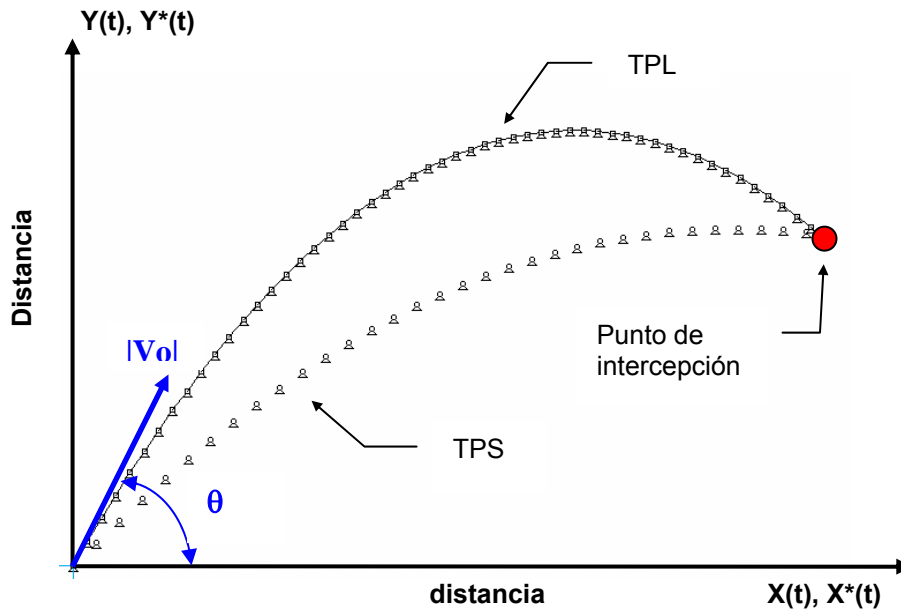
## II.4 ANÁLISIS DEL PROBLEMA DE INTERCEPCIÓN.

Sean dos partículas en movimiento que describen trayectorias TPL (Trayectoria de la Partícula Libre) y TPS (Trayectoria de la Partícula Seguidora) las cuales son eventos espacio-temporales.

Para un intervalo de tiempo determinado  $[0, T^*]$ , TPL inicia su movimiento en el instante  $t = t_0$ , con una velocidad  $V_0$  y un ángulo  $\theta$ , Mientras que TPS se inicia en el tiempo  $t = t_1$  con  $t_0 < t_1 < t_2$ . El problema que se pretende resolver consiste en determinar matemáticamente a TPS la cual tendrá que interceptar a TPL en el instante  $t_2$  que está acotado en  $t_1 < t_2 < T^*$ .

## II.4.1 EL FENÓMENO FÍSICO

La figura 2.1 muestra el fenómeno físico del fenómeno de intercepción de la partícula TPL por la partícula TPS en un punto de intercepción predefinido.



**Figura 2.1.** Fenómeno físico de intercepción.

TPL.- Trayectoria de Partícula Libre.

TPS.- Trayectoria de Partícula Seguidora.

La representación paramétrica del movimiento de la partícula TPL se caracteriza por medio de las funciones  $X(t)$  y  $Y(t)$  mostradas a continuación: Es importante mencionar que las funciones en el tiempo pueden ser de cualquier tipo.

La posición en el tiempo está dada por la ecuación (2.1)

$$\begin{aligned} X(t) &= |V_0| \cos \theta t \\ Y(t) &= |V_0| \sin \theta t - \frac{1}{2}gt^2 \end{aligned} \quad (2.1)$$

La velocidad se representa por la ecuación (2.2).

$$\begin{aligned} X(t) &= |V_0| \cos \theta \\ Y(t) &= |V_0| \sin \theta - gt \end{aligned} \quad (2.2)$$

La aceleración por la ecuación (2.3)

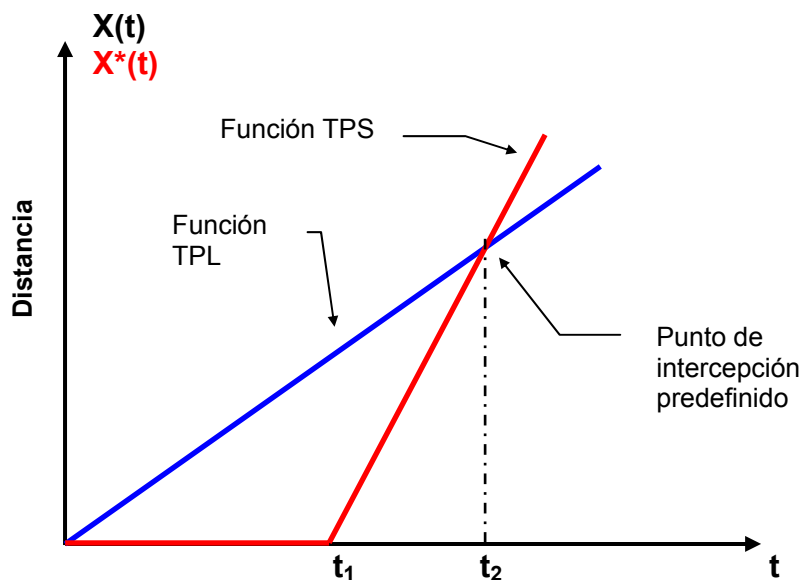
$$\begin{aligned} \dot{X}(t) &= 0 \\ \dot{Y}(t) &= -g \end{aligned} \quad (2.3)$$

Cabe señalar que las ecuaciones 2.1, 2.2 y 2.3 están representadas en el sistema de coordenadas cartesiano  $\mathbf{X}(t)$  y  $\mathbf{Y}(t)$ . Además, se observa que la función  $\mathbf{X}(t)$  genera un movimiento rectilíneo uniforme y la función  $\mathbf{Y}(t)$  un movimiento uniformemente acelerado. Así se tiene, que las funciones  $\mathbf{X}(t):[0, T] \rightarrow \mathbb{R}^+$  y  $\mathbf{Y}(t):[0, T] \rightarrow \mathbb{R}$  están representadas por las figuras 2.2, 2.3, 2.4 mostradas a continuación:

#### II.4.2. EL FENÓMENO REAL.

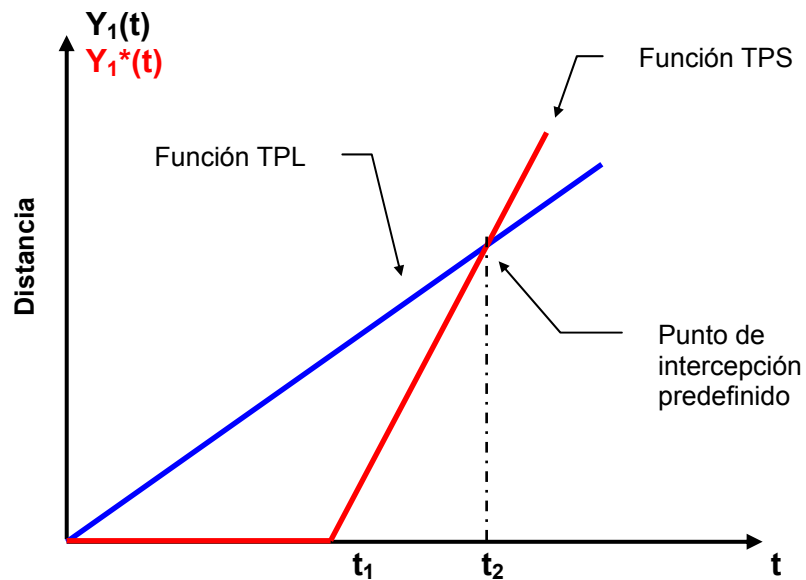
El fenómeno real de intercepción de trayectorias involucra al espacio-tiempo. Se puede observar en las figuras 2.2, 2.3 y 2.4 que las funciones  $\mathbf{X}(t):[t, T] \rightarrow \mathbb{R}^+$ ,  $\mathbf{Y}(t) = \mathbf{Y}_1(t) + \mathbf{Y}_2(t)$  donde  $\mathbf{Y}_1(t):[0, T] \rightarrow \mathbb{R}^+$  y  $\mathbf{Y}_2(t):[0, T] \rightarrow \mathbb{R}^-$ .

El objetivo es determinar las funciones  $\mathbf{X}^*(t):[t_1, T^*] \rightarrow \mathbb{R}^+$ ,  $\mathbf{Y}_1^*(t):[t_1, T^*] \rightarrow \mathbb{R}^+$  y  $\mathbf{Y}_2^*(t):[t_1, T^*] \rightarrow \mathbb{R}^-$  con  $T^* \leq T$  de tal manera que con un tiempo de retardo:  $t = t_1 \in \mathbb{R}^+$  en el tiempo, intercepte las funciones  $\mathbf{X}(t)$ ,  $\mathbf{Y}_1(t)$  y  $\mathbf{Y}_2(t)$  en el tiempo de intercepción  $t = t_2 \in \mathbb{R}^+$ . Para ello, las funciones que definen la trayectoria TPS tienen un comportamiento como el mostrado en las figuras 2.2, 2.3 Y 2.4.



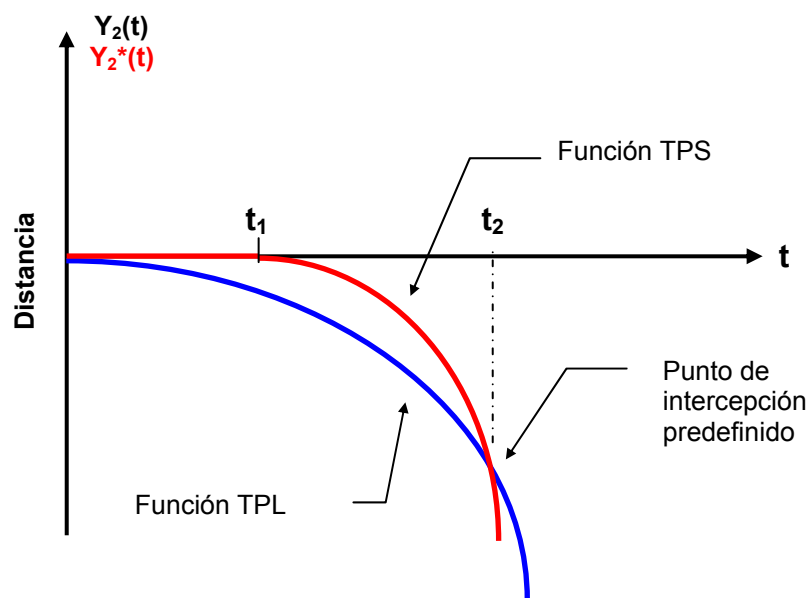
**Figura 2.2** La función TPL está definida sobre el sistema coordenado  $(\mathbf{X}(t), t)$ , y la función TPS en  $(\mathbf{X}^*(t), t)$ .

La figura 2.2 muestra que las funciones  $X(t)$  y  $X^*(t)$  son lineales.



**Figura 2.3** La función TPL está definida sobre el sistema coordinado  $(Y_1(t), t)$ , y la función TPS en  $(Y_1^*(t), t)$ .

La función de la figura 2.3 tiene un comportamiento lineal y la función de la figura 2.4 un comportamiento no lineal.



**Figura 2.4** La función TPL está definida sobre el sistema coordinado  $(Y_2(t), t)$ , y la función TPS en  $(Y_2^*(t), t)$ .

Además, en las figuras 2.2, 2.3 y 2.4 se asocia el sistema coordinado  $(X^*(t), t)$  móvil haciéndolo coincidir en el origen y con los ejes paralelos al sistema coordinado  $(X(t), t)$  fijo.

## II.5 PLANTEAMIENTO DEL MODELO [2.5]

La definición de la función seguidora TPS formada por las siguientes funciones:

$$X^*(t), \quad Y^*(t) = Y_1^*(t) + Y_2^*(t) \quad (2.4)$$

se hará de acuerdo con la siguiente metodología. Incisos i), ii), iii).

i) Observando las figuras 2.2, 2.3 y 2.4 se plantean las siguientes restricciones:

$$X^*(t) = Y_1^*(t) = Y_2^*(t) = 0, \quad 0 \leq t \leq t_1 \quad (2.5)$$

ii) Las funciones  $X^*(t):[t_1, T^*] \rightarrow \mathcal{R}^+$  y  $Y_1^*(t):[t_1, T^*] \rightarrow \mathcal{R}^+$  son lineales, mientras que la función  $Y_2^*(t):[t_1, T^*]$  es una función cuadrática. Además, como se considera que los sistemas coordenados coinciden en el origen y están orientados paralelamente:

$$\begin{aligned} X^*(t_2) &= X(t_2) \\ Y_1^*(t_2) &= Y_1(t_2) \\ Y_2^*(t_2) &= Y_2(t_2), \quad t_1 < t_2 \leq T^* \end{aligned} \quad (2.6)$$

iii) De acuerdo a i) y ii), se puede proponer la forma de la función TPS buscada planteando el siguiente sistema de ecuaciones:

$$\begin{aligned} X^*(t) &= \alpha_0 + \alpha_1 t \\ Y_1^*(t) &= \beta_0 + \beta_1 t \\ Y_2^*(t) &= \gamma_0 + \gamma_1 t + \gamma_2 t^2, \quad t \in [t_1, T^*] \end{aligned} \quad (2.7)$$

## II.6 SOLUCIÓN AL CASO DE LA TRAYECTORIA DEL TIRO PARABÓLICO

### II.6.1 ANÁLISIS DE POSICIÓN

Aplicando la metodología antes descrita al caso planteado de la trayectoria descrita por el fenómeno del tiro parabólico, según i), ii) y iii) se determinan los coeficientes de la siguiente forma:

$$a) \quad X^*(t) = \alpha_0 + \alpha_1 t \quad (2.8)$$

$$X^*(t_1) = \alpha_0 + \alpha_1 t_1 = 0$$

$$X^*(t_2) = \alpha_0 + \alpha_1 t_2 = |V_0| \cos \theta t_2$$

Resolviendo el sistema anterior, se obtiene:

$$\alpha_1 = \frac{1}{t_2 - t_1} |V_0| \text{Cos} \theta t_2$$

$$\alpha_0 = -\frac{t_1}{t_2 - t_1} |V_0| \text{Cos} \theta t_2$$

sustituyendo lo anterior en (2.8) se obtiene:

$$\begin{aligned} X^*(t) &= 0, & 0 \leq t \leq t_1 \\ X^*(t) &= \frac{1}{t_2 - t_1} |V_0| \text{Cos} \theta t_2 (t - t_1), & t > t_1 \end{aligned} \quad (2.9)$$

La función  $Y^*(t)$  se divide en las funciones  $Y_1^*(t)$  y  $Y_2^*(t)$ , y se evalúa en forma individual a cada una de ellas y después se suman para obtener la función  $Y^*(t)$ .

**b)  $Y_1^*(t) = \beta_0 + \beta_1 t$**

Aplicando la misma formulación del inciso a), ecuación 2.8, se tiene:

$$Y_1^*(t_1) = \beta_0 + \beta_1 t_1 = 0$$

$$Y_1^*(t_2) = \beta_0 + \beta_2 t_2 = |V_0| \text{Sen} \theta t_2$$

Resolviendo el sistema, se obtiene:

$$\begin{aligned} Y_1^*(t) &= 0, & 0 \leq t \leq t_1 \\ Y_1^*(t) &= \frac{1}{t_2 - t_1} |V_0| \text{Sen} \theta t_2 (t - t_1), & t > t_1 \end{aligned} \quad (2.10)$$

**c)  $Y_2^*(t) = \gamma_0 + \gamma_1 t + \gamma_2 t^2$**

$$Y_2^*(t_1) = \gamma_0 + \gamma_1 t_1 + \gamma_2 t_1^2 = 0$$

$$Y_2^*(t_2) = \gamma_0 + \gamma_1 t_2 + \gamma_2 t_2^2 = -\frac{1}{2} g t_2^2 \quad (2.11)$$

De la restricción i) y derivando  $Y_2^*(t)$  con respecto al tiempo, para obtener dos ecuaciones lineales y completar el sistema de ecuaciones de 3x3 de las ecuaciones mostradas en (2.11), se tiene que:

$$\frac{\partial}{\partial t} Y_2^*(t) = \gamma_1 + 2 \gamma_2 t = 0 \quad (2.12)$$

Tomando los límites de  $t$ . Es decir,  $t = 0$  y  $t = t_1$  se presentan los casos siguientes:

**Caso I (t = 0)**

$\frac{\partial}{\partial t} Y_2^*(0) = \gamma_1 = 0$  y con las ecuaciones 2.11 inciso c),

$$\begin{aligned} \gamma_1 &= 0 \\ Y_2^*(t_1) &= \gamma_0 + \gamma_1 t_1 + \gamma_2 t_1^2 = 0 \\ Y_2^*(t_2) &= \gamma_0 + \gamma_1 t_2 + \gamma_2 t_2^2 = -\frac{1}{2} g t_2^2 \end{aligned} \quad (2.13)$$

Resolviendo el sistema anterior, se obtiene:

$Y_2^*(t) = 0,$	$0 \leq t \leq t_1$
$Y_2^*(t) = -\frac{g t_2^2}{2(t_2^2 - t_1^2)}(t^2 - t_1^2),$	$t > t_1$

Y recordando que  $Y^*(t) = Y_1^*(t) + Y_2^*(t)$ , se obtiene el caso I

$Y^*(t) = 0,$	$0 \leq t \leq t_1$
$Y^*(t) = \frac{1}{t_2 - t_1}  V_0  \text{Sen } \theta t_2 (t - t_1) - \frac{g t_2^2}{2(t_2^2 - t_1^2)}(t^2 - t_1^2),$	$t > t_1$

**(2.14)**

**Caso II (t = t<sub>1</sub>)**

$\frac{\partial}{\partial t} Y_2^*(t_1) = \gamma_1 + 2\gamma_2 t_1 = 0$

utilizando las ecuaciones (2.11) se forma el siguiente sistema de ecuaciones de 3x3.

$$\begin{aligned} \gamma_1 + 2\gamma_2 t_1 &= 0 \\ Y_2^*(t_1) &= \gamma_0 + \gamma_1 t_1 + \gamma_2 t_1^2 = 0 \\ Y_2^*(t_2) &= \gamma_0 + \gamma_1 t_2 + \gamma_2 t_2^2 = -\frac{1}{2} g t_2^2 \end{aligned}$$

Resolviendo el sistema anterior, obtenemos:

$Y_2^*(t) = 0,$	$0 \leq t \leq t_1$
$Y_2^*(t) = -\frac{g t_2^2}{2(t_2 - t_1)^2}(t - t_1)^2,$	$t > t_1$

Finalmente se tiene:



$$\begin{aligned}
 Y^*(t) &= 0, & 0 \leq t \leq t_1 \\
 Y^*(t) &= \frac{1}{t_2 - t_1} |V_0| \text{Sen } \theta t_2 (t - t_1) - \frac{g t_2^2}{2(t_2 - t_1)^2} (t - t_1)^2, & t > t_1
 \end{aligned} \tag{2.15}$$

### II.6.2 ANÁLISIS DE VELOCIDAD

Para encontrar la función de velocidad de las dos funciones TPL y TPS se necesita aplicar la derivada parcial con respecto a t.

Para la partícula libre TPL: se toman como constantes  $|V_0|$ ,  $\theta$ , g

$$\begin{aligned}
 V_x(t) &= \frac{\partial}{\partial t} X(t) = |V_0| \text{Cos } \theta \\
 V_y(t) &= \frac{\partial}{\partial t} Y(t) = |V_0| \text{Sen } \theta - g t
 \end{aligned}$$

Para la partícula seguidora TPS: se toman como constantes  $|V_0|$ ,  $\theta$ , g,  $t_1$ ,  $t_2$ ,

$$\begin{aligned}
 V_x^*(t) &= \frac{\partial}{\partial t} X^*(t) = -\frac{1}{t_1 - t_2} = |V_0| \text{Cos } \theta t_2 \\
 V_y^*(t) &= \frac{\partial}{\partial t} Y^*(t) = \frac{1}{t_1 - t_2} = |V_0| \text{Sen } \theta t_2 + \frac{g t_2^2 t}{t_1^2 - t_2^2}
 \end{aligned}$$

### II.6.3 ANÁLISIS DE ACELERACIÓN

Para encontrar la función de aceleración de las dos funciones TPL y TPS se necesita aplicar la derivada parcial nuevamente pero ahora a las funciones de velocidad encontradas anteriormente, con respecto a t.

Para la partícula Libre TPL: se toman como constantes  $|V_0|$ ,  $\theta$ , g.

$$\begin{aligned}
 A_x &= \frac{\partial}{\partial t} V_x(t) = 0 \\
 A_y &= \frac{\partial}{\partial t} V_y(t) = \frac{g t_2^2}{t_1^2 - t_2^2}
 \end{aligned}$$

Para la partícula segadora TPS: se toman como constantes  $|V_0|$ ,  $\theta$ , g,  $t_1$ ,  $t_2$ .

$$A_x^* = \frac{\partial}{\partial t} V_x^*(t) = 0$$

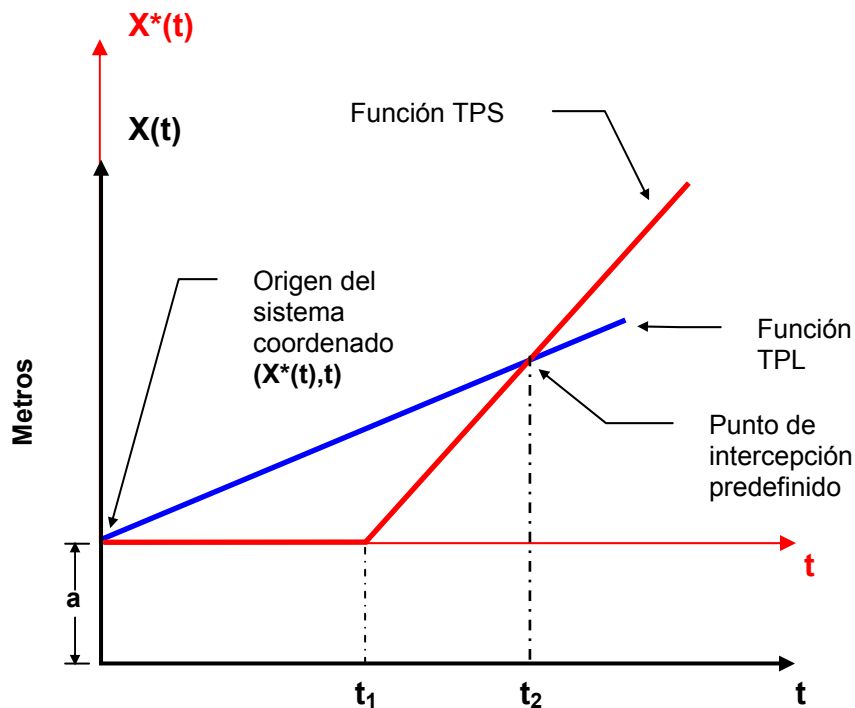
$$Ay^* = \frac{\partial}{\partial t} Vy^*(t) = \frac{g t_2^2}{t_1^2 - t_2^2}$$

Se puede observar que la aceleración de la partícula seguidora es constante y no depende del tiempo absoluto, si no de los siguientes valores; tiempo de intercepción, tiempo de retardo y de la aceleración, en este caso la provocada por la gravedad.

## II.7 CAMBIO DE BASE

Observando el evento de intercepción se puede observar que los sistemas coordenados coinciden en el origen. A veces es necesario cambiar la posición de uno de los sistemas coordenados, para lograrlo se realizará la siguiente modificación:

Se desplazará el sistema de coordenadas de  $(X^*(t), t)$  una distancia  $+a$  unidades como a continuación se muestra en la figura 2.5. La modificación a la metodología de estudio que se planteó anteriormente representará un desplazamiento a lo largo del eje  $X(t)$  en el fenómeno físico.



**Figura 2.5.** Representación gráfica del cambio de base para el fenómeno de intercepción.

### II.7.1 ANÁLISIS DE POSICIÓN

Aplicando la metodología vista anteriormente:

$$\mathbf{a)} \quad \mathbf{X}^*(\mathbf{t}) = \alpha_0 + \alpha_1 \mathbf{t}$$

$$\mathbf{X}^*(\mathbf{t}_1) = \alpha_0 + \alpha_1 \mathbf{t}_1 = \mathbf{a}$$

$$\mathbf{X}^*(\mathbf{t}_2) = \alpha_0 + \alpha_1 \mathbf{t}_2 = |\mathbf{V}_0| \mathbf{C} \cos \theta \mathbf{t}_2$$

Resolviendo el sistema anterior y sustituyendo en **a)**, se obtiene:

$$\begin{aligned} \mathbf{X}^*(\mathbf{t}) &= \mathbf{0}, & 0 \leq \mathbf{t} \leq \mathbf{t}_1 \\ \mathbf{X}^*(\mathbf{t}) &= \frac{1}{\mathbf{t}_2 - \mathbf{t}_1} \left( |\mathbf{V}_0| \mathbf{C} \cos \theta \mathbf{t}_2 (\mathbf{t} - \mathbf{t}_1) - \mathbf{a} (\mathbf{t} - \mathbf{t}_2) \right), & \mathbf{t} > \mathbf{t}_1 \end{aligned} \quad (2.16)$$

Como sólo se ha desplazado el sistema de coordenadas ( $\mathbf{X}^*(\mathbf{t}), \mathbf{t}$ ), ahora se realizará el análisis cinemático:

### II.7.2 ANÁLISIS DE VELOCIDAD Y ACELERACIÓN

Para la partícula seguidora TPS: tomando como constantes  $|\mathbf{V}_0|, g, \theta, \mathbf{t}_1, \mathbf{t}_2, \mathbf{a}$ .

$$\mathbf{Vx}^*(\mathbf{t}) = \frac{\partial}{\partial \mathbf{t}} \mathbf{X}^*(\mathbf{t}) = \frac{1}{\mathbf{t}_1 - \mathbf{t}_2} \left( \mathbf{a} - |\mathbf{V}_0| \mathbf{C} \cos \theta \mathbf{t}_2 \right)$$

Para la partícula seguidora TPS: tomando como constantes  $|\mathbf{V}_0|, g, \theta, \mathbf{t}_1, \mathbf{t}_2, \mathbf{a}$ .

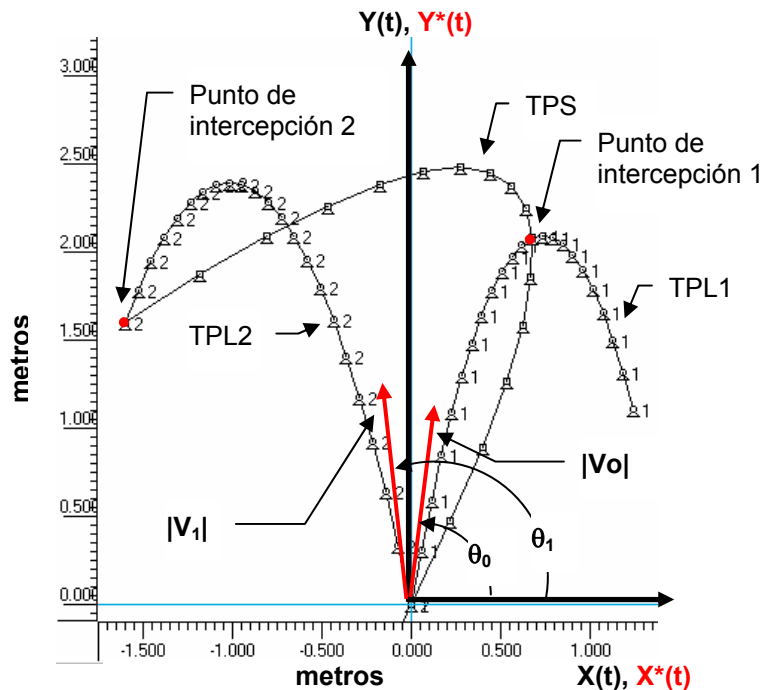
$$\mathbf{Ax}^* = \frac{\partial}{\partial \mathbf{t}} \mathbf{Vx}^*(\mathbf{t}) = \mathbf{0}$$

### II.8 INTERCEPCIÓN MÚLTIPLE DE DOS PARTICULAS

Para generar las funciones que intercepten dos partículas con velocidades y ángulos de disparo diferentes, se utilizará la metodología antes vista y se modificará lo siguiente, lo cual se muestra en las figuras 2.6, 2.7 y 2.8. Se ha añadido la función de la trayectoria de otra partícula. Por lo tanto, ahora la partícula 1 es llamada TPL1 y la partícula 2 TPL2, con estas funciones se planteará el sistema de ecuaciones necesario para resolver el problema de intercepción múltiple.

## II.8.1 EL FENÓMENO REAL.

Para definir la función de la trayectoria seguidora TPS, se considerará lo siguiente:



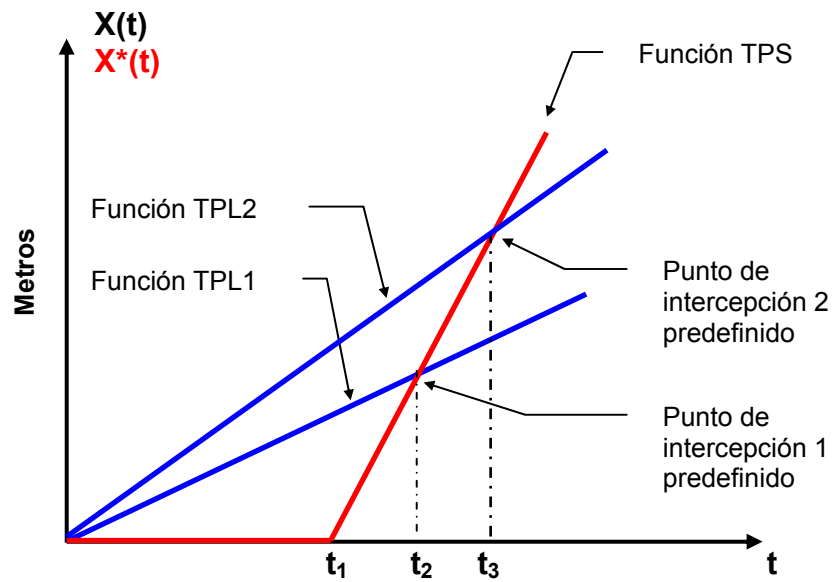
**Figura 2.6** Fenómeno físico de intercepción múltiple de dos partículas; donde TPL1.- es la trayectoria de la partícula libre 1 y TPL2.- la trayectoria de la partícula 2. TPS.- Trayectoria de la partícula seguidora.

En la figura 2.6, se muestra la idea general de la intercepción múltiple de trayectorias. En este caso, hay dos trayectorias llamadas TPL1 Y TPL2 que son las partículas libres a interceptar.

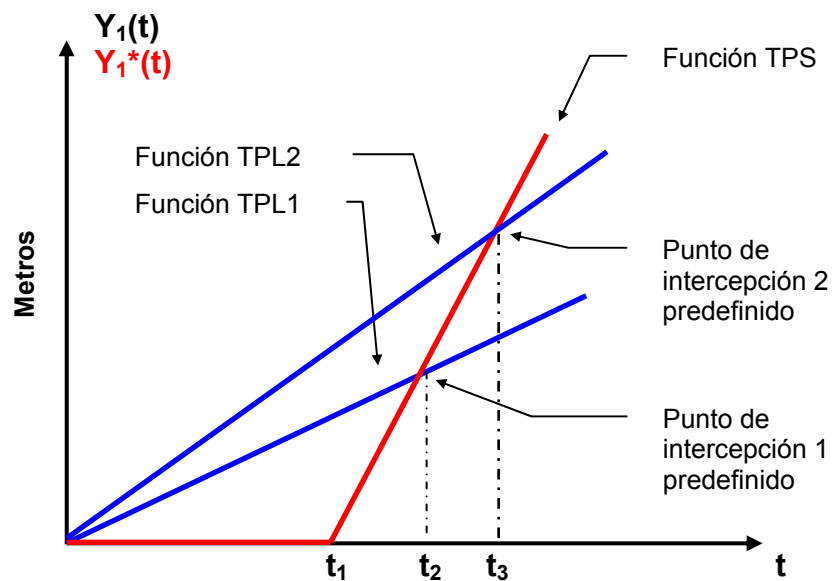
La velocidad inicial y el ángulo de disparo para la trayectoria TPL1 son:  $|V_0|$  y  $\theta_0$  y para la trayectoria TPL2, velocidad inicial  $|V_1|$  y ángulo de disparo  $\theta_1$ . La trayectoria seguidora TPS intercepta en el punto de intercepción 1 a la trayectoria TPL1 y luego intercepta la segunda trayectoria TPL2 en el punto de intercepción 2. El tiempo de intercepción será independiente para cada trayectoria.

La trayectoria TPS debe calcularse simultáneamente para interceptar las dos partículas. Por lo tanto, el sistema de ecuaciones debe plantearse considerando ahora dos partículas libres y los tiempos de intercepción serán  $t_2$  y  $t_3$  respectivamente.

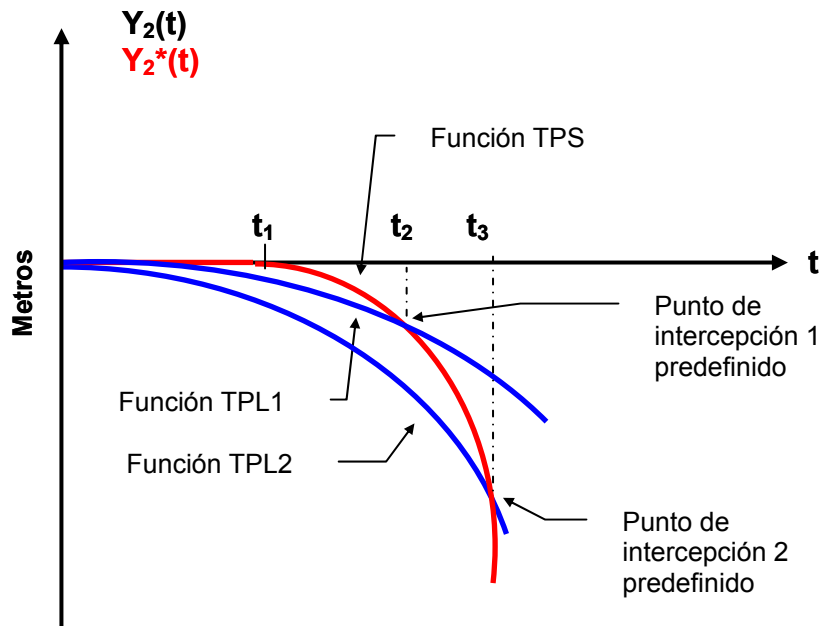
Las figuras 2.7, 2.8 Y 2.9 incluyen las funciones de las partículas libres y los tiempos de intercepción y se procede de la misma forma utilizando la metodología planteada.



**Figura 2.7** Las funciones TPL1 y TPL2 están definidas sobre el sistema coordenado  $(X(t),t)$ , y la función TPS en  $(X^*(t),t)$ .



**Figura 2.8** Las funciones TPL1 y TPL2 están definidas sobre el sistema coordenado  $(Y_1(t),t)$ , y la función TPS en  $(Y_1^*(t),t)$ .



**Figura 2.9** Las funciones TPL1 y TPL2 están definidas sobre el sistema coordenado  $(Y_2(t), t)$ , y la función TPS en  $(Y_2^*(t), t)$ .

## II.8.2 ANÁLISIS DE POSICIÓN

Aplicando la metodología antes descrita al caso planteado de la trayectoria descrita por el fenómeno del tiro parabólico, según i), ii) y iii) se determinan los coeficientes de la siguiente forma:

Para la partícula seguidora TPS:

$$\mathbf{a)} \quad \mathbf{X}^*(t) = \alpha_0 + \alpha_1 t + \alpha_2 t^2$$

Se Plantea el siguiente sistema de ecuaciones:

$$\mathbf{X}^*(t_1) = \alpha_0 + \alpha_1 t_1 + \alpha_2 t_1^2 = 0$$

$$\mathbf{X}^*(t_2) = \alpha_0 + \alpha_1 t_2 + \alpha_2 t_2^2 = |\mathbf{V}_0| \cos \theta_0 t_2$$

$$\mathbf{X}^*(t_3) = \alpha_0 + \alpha_1 t_3 + \alpha_2 t_3^2 = |\mathbf{V}_1| \cos \theta_1 t_3$$

Resolviendo el sistema anterior:

$$\mathbf{X}^*(t) = 0, \quad 0 \leq t \leq t_1$$

$$\mathbf{X}^*(t) = a((t - t_1)(t_2 t_3 b) + t_1(c + t_2 d)), \quad t \geq t_1$$

Donde;

$$a = -\frac{1}{(t_1 - t_2)(t_1 - t_3)(t_2 - t_3)}$$

$$b = -|V_0| \cos \theta_0 (t - t_3) + |V_1| \cos \theta_1 (t - t_2)$$

$$c = -|V_1| \cos \theta_1 t_3 t$$

$$d = |V_0| \cos \theta_0 (t - t_3) + |V_1| \cos \theta_1 t_3$$

$$\mathbf{b) \quad Y_1^*(t) = \beta_0 + \beta_1 t + \beta_2 t^2}$$

Planteamos el siguiente sistema de ecuaciones:

$$Y_1^*(t_1) = \beta_0 + \beta_1 t_1 + \beta_2 t_1^2 = 0$$

$$Y_1^*(t_2) = \beta_0 + \beta_1 t_2 + \beta_2 t_2^2 = |V_0| \operatorname{Sen} \theta_0 t_2$$

$$Y_1^*(t_3) = \beta_0 + \beta_1 t_3 + \beta_2 t_3^2 = |V_1| \operatorname{Sen} \theta_1 t_3$$

Resolviendo el sistema anterior:

$Y_1^*(t) = 0,$	$0 \leq t \leq t_1$
$Y_1^*(t) = a((t - t_1)(t_2 t_3 b) + t_1(c + t_2 d)), \quad t \geq t_1$	

Donde;

$$a = -\frac{1}{(t_1 - t_2)(t_1 - t_3)(t_2 - t_3)}$$

$$b = -|V_0| \operatorname{Sen} \theta_0 (t - t_3) + |V_1| \operatorname{Sen} \theta_1 (t - t_2)$$

$$c = -|V_1| \operatorname{Sen} \theta_1 t_3 t$$

$$d = |V_0| \operatorname{Sen} \theta_0 (t - t_3) + |V_1| \operatorname{Sen} \theta_1 t_3$$

$$\mathbf{c) \quad Y_2^*(t) = \gamma_0 + \gamma_1 t + \gamma_2 t^2}$$

se obtiene el siguiente sistema de ecuaciones:

$$Y_2^*(t_1) = \gamma_0 + \gamma_1 t_1 + \gamma_2 t_1^2 = 0$$

$$Y_2^*(t_2) = \gamma_0 + \gamma_1 t_2 + \gamma_2 t_2^2 = -\frac{1}{2} g t_2^2$$

$$Y_2^*(t_3) = \gamma_0 + \gamma_1 t_3 + \gamma_2 t_3^2 = -\frac{1}{2} g t_3^2$$

Resolviendo el sistema anterior y haciendo  $Y^*(t) = Y_1^*(t) + Y_2^*(t)$ :

$$Y^*(t) = 0, \quad 0 \leq t \leq t_1$$

$$Y^*(t) = Y_1^*(t) + \frac{g(t-t_1)(-t t_2 t_3 + t_1(t_2(t-t_3) + t t_3))}{2(t_1-t_2)(t_1-t_3)}, \quad t > t_1$$

### II.8.3 ANÁLISIS DE VELOCIDAD

Para encontrar la función de velocidad de las dos partículas TPL y TPS se derivará parcialmente con respecto a t. Para la partícula seguidora TPS: tomando como constantes  $|V_0|$ ,  $\theta_0$ , g,  $|V_1|$ ,  $\theta_1$ ,  $t_1$ ,  $t_2$ ,  $t_3$ .

$V_x^*(t) = \frac{\partial}{\partial t} X(t) = 0, \quad 0 \leq t \leq t_1$
$V_x^*(t) = \frac{\partial}{\partial t} X(t) = a(t_2 t_3 b - 2t t_1 c + t_1^2 d), \quad t > t_1$

Donde;

$$a = \frac{1}{(t_1 - t_2)(t_1 - t_3)(t_2 - t_3)}$$

$$b = |V_0| \cos \theta_0 (2t - t_3) + |V_1| \cos \theta_1 (-2t + t_2)$$

$$c = |V_0| \cos \theta_0 t_2 - |V_1| \cos \theta_1 t_3$$

$$d = |V_0| \cos \theta_0 t_2 - |V_1| \cos \theta_1 t_3$$

$V_y^*(t) = \frac{\partial}{\partial t} Y(t) = 0, \quad 0 \leq t \leq t_1$
$V_y^*(t) = \frac{\partial}{\partial t} Y(t) = a(t_1^2 b + 2t_2 t_3 (c + t_2 d) + 2t t_1 e), \quad t > t_1$

Donde;

$$a = \frac{1}{2(t_1 - t_2)(t_1 - t_3)(t_2 - t_3)}$$

$$b = -gt_2^2 + 2|V_0| \sin \theta_0 t_2 + t_3(gt_3 - 2|V_1| \sin \theta_1)$$

$$c = g t t_3 + 2|V_0| \sin \theta_0 t - |V_0| \sin \theta_0 t_3 - 2|V_1| \sin \theta_1 t$$

$$d = -gt + |V_1| \sin \theta_1$$

$$e = gt_2^2 - 2|V_0| \sin \theta_0 t_2 + t_3(-gt_3 + 2|V_1| \sin \theta_1)$$



## II.8.4 ANÁLISIS DE ACELERACIÓN

Para encontrar la función de aceleración de las dos partículas TPL1, TPL2 Y TPS se derivará nuevamente, pero ahora se aplicará a las funciones de velocidad encontradas anteriormente, respecto  $t$ .

Para la partícula seguidora TPS: tomando como constantes  $|V_0|$ ,  $\theta_0$ ,  $g$ ,  $|V_1|$ ,  $\theta_1$ ,  $t_1$ ,  $t_2$ ,  $t_3$ .

$$\begin{aligned} \mathbf{Ax}^* &= \frac{\partial}{\partial t} \mathbf{Vx}(t) = \mathbf{0}, & 0 \leq t \leq t_1 \\ \mathbf{Ax}^* &= \frac{\partial}{\partial t} \mathbf{Vx}(t) = a(2t_2t_3b + t_1c), & t > t_1 \end{aligned}$$

Donde;

$$\begin{aligned} a &= \frac{1}{(t_1 - t_2)(t_1 - t_3)(t_2 - t_3)} \\ b &= |V_0| \cos \theta_0 - |V_1| \cos \theta_1 \\ c &= -2|V_0| \cos \theta_0 t_2 + 2|V_1| \cos \theta_1 t_3 \end{aligned}$$

$$\begin{aligned} \mathbf{Ay}^* &= \frac{\partial}{\partial t} \mathbf{Vy}(t) = \mathbf{0}, & 0 \leq t \leq t_1 \\ \mathbf{Ay}^* &= \frac{\partial}{\partial t} \mathbf{Vy}(t) = a(t_2t_3b + t_1c), & t > t_1 \end{aligned}$$

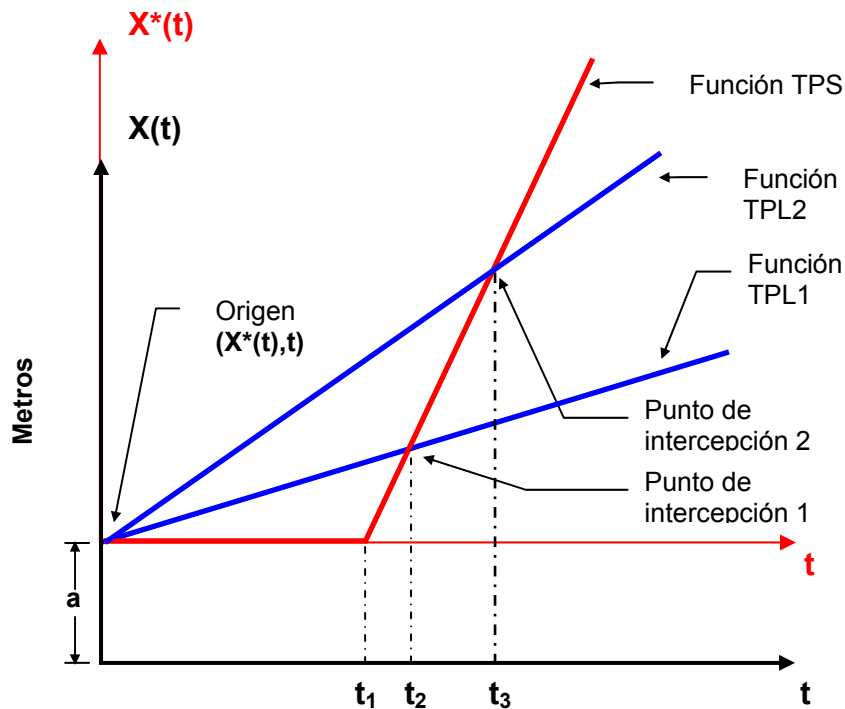
Donde;

$$\begin{aligned} a &= \frac{1}{(t_1 - t_2)(t_1 - t_3)(t_2 - t_3)} \\ b &= -gt_2 + gt_3 + 2|V_0| \sin \theta_0 - 2|V_1| \sin \theta_1 \\ c &= gt_2^2 - 2|V_0| \sin \theta_0 t_2 + t_3(-gt_3 + 2|V_1| \sin \theta_1) \end{aligned}$$

## II.9 CAMBIO DE BASE DOS PARTICULAS

Observando el evento de intercepción se puede ver que los sistemas coordenados coinciden en el origen. A veces es necesario cambiar la posición de uno de los sistemas coordenados. Para lograrlo, se hace la siguiente modificación a la metodología planteada:

Desplazando el sistema de coordenadas de  $(X^*(t), t)$  una distancia  $+a$  unidades, como a continuación se muestra en la figura 2.10. Esto representará un desplazamiento a lo largo del eje  $X(t)$  en el fenómeno físico.



**Figura 2.10** Representación gráfica del cambio de base para el fenómeno de intercepción.

### II.9.1 ANÁLISIS DE POSICIÓN

Aplicando la metodología propuesta al caso planteado de la trayectoria descrita por el fenómeno del tiro parabólico, según i), ii) y iii) se determinarán los coeficientes de la siguiente forma:

Para la partícula seguidora TPS:

$$a) \quad X^*(t) = \alpha_0 + \alpha_1 t + \alpha_2 t^2$$

Planteando el siguiente sistema de ecuaciones:

$$X^*(t_1) = \alpha_0 + \alpha_1 t_1 + \alpha_2 t_1^2 = a$$

$$X^*(t_2) = \alpha_0 + \alpha_1 t_2 + \alpha_2 t_2^2 = |V_0| \cos \theta_0 t_2$$

$$X^*(t_3) = \alpha_0 + \alpha_1 t_3 + \alpha_2 t_3^2 = |V_1| \cos \theta_1 t_3$$

Resolviendo el sistema anterior:

$$\begin{aligned} X^*(t) &= a, & 0 \leq t \leq t_1 \\ X^*(t) &= a(tt_3b + t_2(c + td + e) + t_2^2f), & t \geq t_1 \end{aligned}$$

Donde;

$$a = \frac{1}{(t_1 - t_2)(t_1 - t_3)(t_2 - t_3)}$$

$$b = -at + at_3 + |V_1| \cos \theta_1 t t_1 - |V_1| \cos \theta_1 t_1^2$$

$$c = -t_3^2(a + |V_0| \cos \theta_0 (t - t_1))$$

$$d = at - |V_0| \cos \theta_0 t t_1 + |V_0| \cos \theta_0 t_1^2$$

$$e = (t^2 - t_1^2) t_3 (|V_0| \cos \theta_0 - |V_1| \cos \theta_1)$$

$$f = -at + t_3(a + |V_1| \cos \theta_1 (t - t_1))$$

$$b) \quad Y_1^*(t) = \beta_0 + \beta_1 t + \beta_2 t^2$$

Obteniéndose el siguiente sistema de ecuaciones:

$$Y_1^*(t_1) = \beta_0 + \beta_1 t_1 + \beta_2 t_1^2 = b$$

$$Y_1^*(t_2) = \beta_0 + \beta_1 t_2 + \beta_2 t_2^2 = |V_0| \text{Sen } \theta_0 t_2$$

$$Y_1^*(t_3) = \beta_0 + \beta_1 t_3 + \beta_2 t_3^2 = |V_1| \text{Sen } \theta_1 t_3$$

Resolviendo el sistema anterior:

$$\begin{aligned} Y_1^*(t) &= b, & 0 \leq t \leq t_1 \\ Y_1^*(t) &= a(tt_3b + t_2(c + td + e) + t_2^2f), & t \geq t_1 \end{aligned}$$

Donde;

$$a = \frac{1}{(t_1 - t_2)(t_1 - t_3)(t_2 - t_3)}$$

$$b = -bt + bt_3 + |V_1| \text{Sen } \theta_1 t t_1 - |V_1| \text{Sen } \theta_1 t_1^2$$

$$c = -t_3^2(b + |V_0| \text{Sen } \theta_0 (t - t_1))$$

$$d = bt - |V_0| \text{Sen } \theta_0 t t_1 + |V_0| \text{Sen } \theta_0 t_1^2$$

$$e = (t^2 - t_1^2) t_3 (|V_0| \text{Sen } \theta_0 - |V_1| \text{Sen } \theta_1)$$

$$f = -bt + t_3(b + |V_1| \text{Sen}\theta_1(t - t_1))$$

c)  $Y_2^*(t) = \gamma_0 + \gamma_1 t + \gamma_2 t^2$

Dando origen al siguiente sistema de ecuaciones:

$$Y_2^*(t_1) = \gamma_0 + \gamma_1 t_1 + \gamma_2 t_1^2 = b$$

$$Y_2^*(t_2) = \gamma_0 + \gamma_1 t_2 + \gamma_2 t_2^2 = -\frac{1}{2} g t_2^2$$

$$Y_2^*(t_3) = \gamma_0 + \gamma_1 t_3 + \gamma_2 t_3^2 = -\frac{1}{2} g t_3^2$$

Resolviendo el sistema anterior y  $Y^*(t) = Y_1^*(t) + Y_2^*(t)$ :

$$Y^*(t) = b, \quad 0 \leq t \leq t_1$$

$$Y^*(t) = Y_1^*(t) + a(tb + t_2c), \quad t \geq t_1$$

Donde;

$$a = \frac{1}{2(t_1 - t_2)(t_1 - t_3)}$$

$$b = 2bt + (-2b + gtt_1 - gt_1^2)t_3$$

$$c = -2bt + gt^2t_1 + (2b - gt^2)t_3 + gt_1^2(-t + t_3)$$

### II.9.2 ANÁLISIS DE VELOCIDAD

Para encontrar la función de velocidad de las dos partículas TPL y TPS se deriva parcialmente con respecto a t.

Para la partícula seguidora TPS: tomando como constantes  $|V_0|$ ,  $\theta_0$ , g,  $|V_1|$ ,  $\theta_1$ ,  $t_1$ ,  $t_2$ ,  $t_3$ , a.

$$Vx^*(t) = \frac{\partial}{\partial t} X(t) = 0, \quad 0 \leq t \leq t_1$$

$$Vx^*(t) = \frac{\partial}{\partial t} X(t) = a(t_3b + t_2^2c + t_2(d + e)), \quad t \geq t_1$$

Donde;

$$a = \frac{1}{(t_1 - t_2)(t_1 - t_3)(t_2 - t_3)}$$

$$b = -2at + at_3 + 2|V_1| \text{Cos}\theta_1 t t_1 - |V_1| \text{Cos}\theta_1 t_1^2$$

$$c = -a + |V_1| \text{Cos}\theta_1 t_3$$

$$d = 2at - 2|V_0| \text{Cos}\theta_0 t t_1 + |V_0| \text{Cos}\theta_0 t_1^2 - |V_0| \text{Cos}\theta_0 t_3^2$$

$$e = 2t t_3 (|V_0| \text{Cos}\theta_0 - |V_1| \text{Cos}\theta_1)$$

$$\begin{aligned} \mathbf{V}_y^*(t) &= \frac{\partial}{\partial t} \mathbf{Y}(t) = \mathbf{0}, & 0 \leq t \leq t_1 \\ \mathbf{V}_y^*(t) &= \frac{\partial}{\partial t} \mathbf{Y}(t) = a(-t_2^2 b + 2t_2(c + d + e) + t_3(f - g)), & t \geq t_1 \end{aligned}$$

Donde;

$$a = \frac{1}{2(t_1 - t_2)(t_1 - t_3)(t_2 - t_3)}$$

$$b = 4b - 2gt t_1 + gt_1^2 + 2t_3(gt - |V_1| \text{Sen}\theta_1)$$

$$c = 4bt - 2|V_0| \text{Sen}\theta_0 t t_1 + |V_0| \text{Sen}\theta_0 t_1^2$$

$$d = t_3^2(gt - |V_0| \text{Sen}\theta_0)$$

$$e = 2t t_3 (|V_0| \text{Sen}\theta_0 - |V_1| \text{Sen}\theta_1)$$

$$f = (4b - 2gt t_1 + gt_1^2)t_3$$

$$g = 2(4bt - 2|V_1| \text{Sen}\theta_1 t t_1 + |V_1| \text{Sen}\theta_1 t_1^2)$$

### II.9.3 ANALISIS DE ACELERACIÓN

Para encontrar la función de aceleración de las dos partículas TPL y TPS se deriva parcialmente, pero ahora a las funciones de velocidad encontradas anteriormente, con respecto a  $t$ .

Para la partícula seguidora TPS: tomando como constantes  $|v_0|$ ,  $\theta_0$ ,  $g$ ,  $|v_1|$ ,  $\theta_1$ ,  $t_1$ ,  $t_2$ ,  $t_3$ ,  $a$ .

$$\begin{aligned} \mathbf{A}_x^* &= \frac{\partial}{\partial t} \mathbf{V}_x(t) = \mathbf{0}, & 0 \leq t \leq t_1 \\ \mathbf{A}_x^* &= \frac{\partial}{\partial t} \mathbf{V}_x(t) = a(2(t_3 b + c)), & t \geq t_1 \end{aligned}$$

Donde;

$$a = \frac{1}{(t_1 - t_2)(t_1 - t_3)(t_2 - t_3)}$$

$$b = -a + |V_1| \cos \theta_1 t_1$$

$$c = t_2(a - |V_0| \cos \theta_0 t_1 + t_3(|V_0| \cos \theta_0 - |V_1| \cos \theta_1))$$

$Ay^* = \frac{\partial}{\partial t} Vy(t) = 0, \quad 0 \leq t \leq t_1$
$Ay^* = \frac{\partial}{\partial t} Vy(t) = a(b + t_2(c + d)), \quad t \geq t_1$

Donde;

$$a = \frac{1}{(t_1 - t_2)(t_1 - t_3)(t_2 - t_3)}$$

$$b = gt_2^2(t_1 - t_3) - t_3(4b + t_1(gt_3 - 2|V_1| \sin \theta_1))$$

$$c = 4b + gt_3^2 - 2|V_0| \sin \theta_0 t_1$$

$$d = 2t_3(|V_0| \sin \theta_0 - |V_1| \sin \theta_1)$$

## II.10 GENERALIZACIÓN DE LA INTERCEPCIÓN MÚLTIPLE DEL FENÓMENO DE TIRO PARABÓLICO.

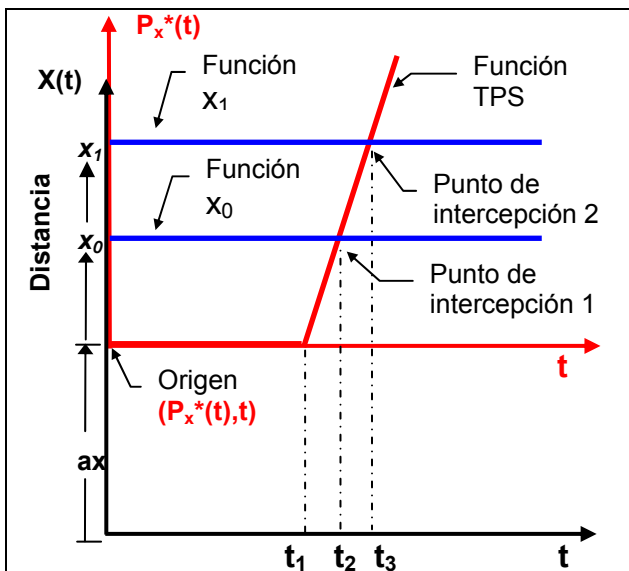
Se presentará el caso general del fenómeno de intercepción múltiple de dos partículas con cambio de base. Un ejemplo de obtención de la trayectoria de la partícula seguidora TPS, utilizando la metodología antes planteada en el capítulo II.

De las ecuaciones paramétricas de las dos partículas TPL1 y TPL2:

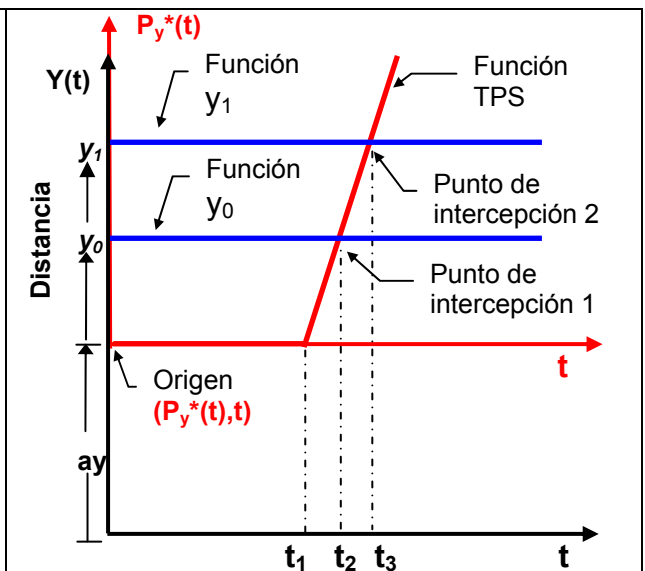
$$\begin{aligned} \text{TPL1: } \quad X(t) &= x_0 + |V_0| \text{Cos } \theta_0 t \\ Y(t) &= y_0 + |V_0| \text{Sen } \theta_0 t - \frac{1}{2}gt^2 \end{aligned} \quad (2.16a)$$

$$\begin{aligned} \text{TPL2: } \quad X(t) &= x_1 + |V_1| \text{Cos } \theta_1 t \\ Y(t) &= y_1 + |V_1| \text{Sen } \theta_1 t - \frac{1}{2}gt^2 \end{aligned} \quad (2.16b)$$

Para encontrar la función seguidora TPS, que intercepte a las partículas libres TPL1 y TPL2 de las ecuaciones (1) y (2) respectivamente, es necesario separarlas de la siguiente forma: ver figura 2.11a a la figura 2.11e.



**Figura 2.11a** Representación gráfica del cambio de base para el fenómeno de intercepción eje X(t).



**Figura 2.11b** Representación gráfica del cambio de base para el fenómeno de intercepción eje Y(t).

En la figura 2.11a se representa la posición “ax” que tendrá el efector final del robot interceptor sobre el eje X(t). Debido a que el robot interceptor se encontrará en una posición siempre fija con respecto al sistema de coordenadas (X(t),t). Por lo tanto, la distancia “ax” será constante durante el desarrollo del evento de intercepción. Por otro lado, las funciones x<sub>0</sub> y x<sub>1</sub> están localizadas a las distancias “ax+x<sub>0</sub>” y “ax+x<sub>0</sub>+x<sub>1</sub>” unidades, respectivamente.

Estas distancias se miden en coordenadas absolutas, Es decir, desde el sistema inercial fijo (X(t),t) y representan sobre el eje X(t) las posiciones iniciales de disparo de las partículas libres TPL1 y TPL2, respectivamente. Es importante mencionar, que las posiciones de las funciones  $x_0$  y  $y_0$  se pueden establecer de manera independiente sobre el eje X(t), como se muestra en la figura 2.11a.

De manera similar, en la figura 2.11b también se representa la posición “ay” que tendrá el efector final del robot interceptor sobre el eje Y(t). Las funciones  $y_0$  y  $y_1$  están localizadas a las distancias “ay+y<sub>0</sub>” y “ay+y<sub>0</sub>+y<sub>1</sub>”, respectivamente. Y representan las posiciones iniciales de disparo de las partículas libres TPL1 y TPL2, respectivamente sobre el eje Y(t). También se miden en coordenadas absolutas desde el eje Y(t).

Cabe mencionar, que las funciones seguidoras  $P_x^*(t)$  y  $P_y^*(t)$  se encuentran definidas en los sistemas móviles ( $P_x^*(t),t$ ) y ( $P_y^*(t),t$ ) y se construyen con las funciones  $x_0, y_0, x_1, y_1$  y con las posiciones  $a_x, a_y$  del sistema fijo (X(t),t) y Y(Y(t),t), respectivamente.

Es importante definir la posición del efector final del robot interceptor en el sistema fijo, para establecer una posición adecuada dependiendo de las condiciones y necesidades del medio físico en donde opere el robot.

A continuación se presentan los sistemas de ecuaciones que permitirán obtener las funciones seguidoras antes mencionadas.

Para construir la función:

$$P_x^*(t) = \alpha_0 + \alpha_1 t + \alpha_2 t^2 \quad t \in [t_1, t_3]$$

los coeficientes se calculan

con el siguiente sistema de ecuaciones:

$$f_1 = \alpha_0 + \alpha_1 t_1 + \alpha_2 t_1^2 = a_x \quad 0 \leq t \leq t_1$$

$$f_2 = \alpha_0 + \alpha_1 t_2 + \alpha_2 t_2^2 = x_0 \quad t_1 < t \leq t_2$$

$$f_3 = \alpha_0 + \alpha_1 t_3 + \alpha_2 t_3^2 = x_1 \quad t_2 < t \leq t_3$$

resolviendo el sistema anterior,

la función  $P_x^*$  resulta de la siguiente forma:

$$P_x^*(t) = p_1(a_x \cdot p_2 - (t - t_1)p_3 + t \cdot p_4),$$

donde;

$$p_1 = \frac{1}{(t_1 - t_2)(t_1 - t_3)(t_2 - t_3)}$$

$$p_2 = (t - t_2)(t - t_3)(t_2 - t_3)$$

$$p_3 = (-t_1 t_3 + t_3^2)x_0 + (t_1 t_2 - t_2^2)x_1$$

$$p_4 = (t_1 - t_3)x_0 - (t_1 - t_2)x_1$$

Para construir la función:

$$P_y^*(t) = \beta_0 + \beta_1 t + \beta_2 t^2 \quad t \in [t_1, t_3]$$

los coeficientes se calculan

con el siguiente sistema de ecuaciones:

$$f_1 = \beta_0 + \beta_1 t_1 + \beta_2 t_1^2 = a_y \quad 0 \leq t \leq t_1$$

$$f_2 = \beta_0 + \beta_1 t_2 + \beta_2 t_2^2 = y_0 \quad t_1 < t \leq t_2$$

$$f_3 = \beta_0 + \beta_1 t_3 + \beta_2 t_3^2 = y_1 \quad t_2 < t \leq t_3$$

resolviendo el sistema anterior,

la función  $P_y^*$  resulta de la siguiente forma:

$$P_y^*(t) = p_1(a_y \cdot p_2 - (t - t_1)p_3 + t \cdot p_4),$$

donde;

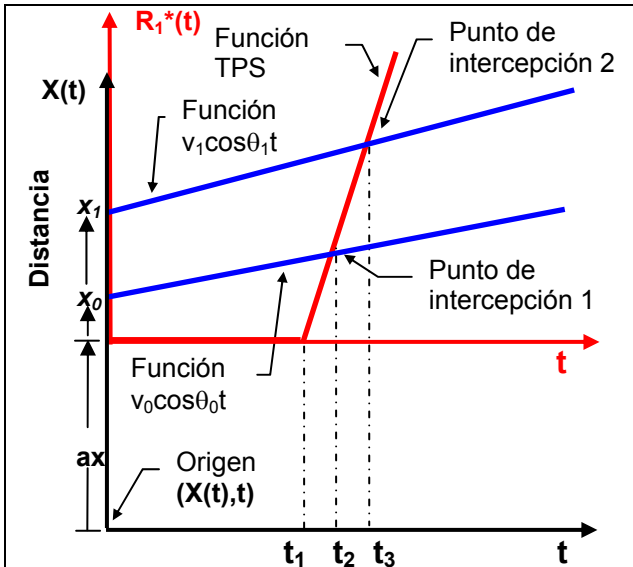
$$p_1 = \frac{1}{(t_1 - t_2)(t_1 - t_3)(t_2 - t_3)}$$

$$p_2 = (t - t_2)(t - t_3)(t_2 - t_3)$$

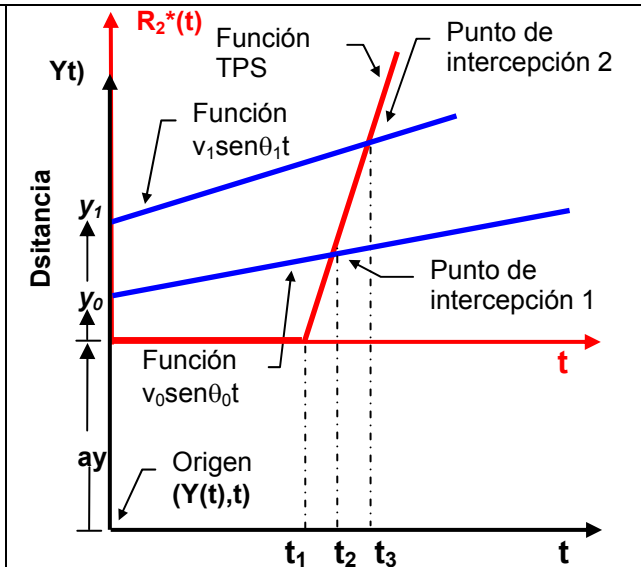
$$p_3 = (-t_1 t_3 + t_3^2)y_0 + (t_1 t_2 - t_2^2)y_1$$

$$p_4 = (t_1 - t_3)y_0 - (t_1 - t_2)y_1$$





**Figura 2.11c** Representación gráfica del cambio de base para el fenómeno de intercepción eje X(t).



**Figura 2.11d** Representación gráfica del cambio de base para el fenómeno de intercepción eje Y(t).

En la figura 2.11c se presentan las funciones  $v_0 \cos \theta_0 t$ ,  $v_1 \cos \theta_1 t$ , tiene un comportamiento lineal, las funciones están desplazadas sobre el eje X(t) una distancia de “ $ax+x_0$ ” y “ $ax+x_0+x_1$ ” unidades, respectivamente. No se considera en este sistema de ecuaciones, debido a que ya está incluido en las funciones  $P_x^*(t)$  y  $P_y^*(t)$ .

Para construir la función:

$$R_1^*(t) = \chi_0 + \chi_1 t + \chi_2 t^2 \quad t \in [t_1, t_3]$$

los coeficientes se calculan con el siguiente sistema de ecuaciones:

$$\begin{aligned} f_1 &= \chi_0 + \chi_1 t_1 + \chi_2 t_1^2 = 0 & 0 \leq t \leq t_1 \\ f_2 &= \chi_0 + \chi_1 t_2 + \chi_2 t_2^2 = v_0 \cos \theta_0 t_2 & t_1 < t \leq t_2 \\ f_3 &= \chi_0 + \chi_1 t_3 + \chi_2 t_3^2 = v_1 \cos \theta_1 t_3 & t_2 < t \leq t_3 \end{aligned}$$

resolviendo el sistema anterior, la función  $R_1^*$  resulta de la siguiente forma:

$$R_1^*(t) = p_1((t - t_1)(t_2(p_2 + p_3))),$$

donde;

$$p_1 = -\frac{1}{(t_1 - t_2)(t_1 - t_3)(t_2 - t_3)}$$

$$p_2 = (t - t_3)(t_1 - t_3)v_0 \cos \theta_0$$

$$p_3 = (t - t_2)(-t_1 + t_2)t_3 v_1 \cos \theta_1$$

Para construir la función:

$$R_2^*(t) = \delta_0 + \delta_1 t + \delta_2 t^2 \quad t \in [t_1, t_3]$$

los coeficientes se calculan con el siguiente sistema de ecuaciones:

$$\begin{aligned} f_1 &= \delta_0 + \delta_1 t_1 + \delta_2 t_1^2 = 0 & 0 \leq t \leq t_1 \\ f_2 &= \delta_0 + \delta_1 t_2 + \delta_2 t_2^2 = v_0 \sin \theta_0 t_2 & t_1 < t \leq t_2 \\ f_3 &= \delta_0 + \delta_1 t_3 + \delta_2 t_3^2 = v_1 \sin \theta_1 t_3 & t_2 < t \leq t_3 \end{aligned}$$

resolviendo el sistema anterior, la función  $R_2^*$  resulta de la siguiente forma:

$$R_2^*(t) = p_1((t - t_1)(t_2(p_2 + p_3))),$$

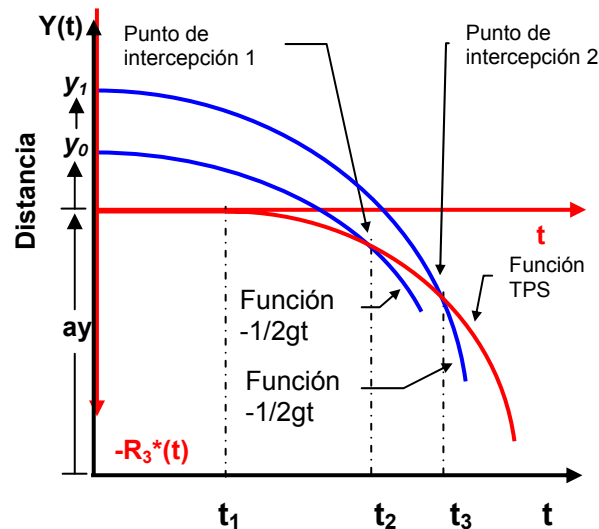
donde;

$$p_1 = -\frac{1}{(t_1 - t_2)(t_1 - t_3)(t_2 - t_3)}$$

$$p_2 = (t - t_3)(t_1 - t_3)v_0 \sin \theta_0$$

$$p_3 = (t - t_2)(-t_1 + t_2)t_3 v_1 \sin \theta_1$$

Para encontrar la función  $R_3^*(t)$ , se plantea la misma metodología:



**Figura 2.11e** Representación gráfica del cambio de base para el fenómeno de intersección eje  $Y(t)$ .

Para construir la función:

$$R_3^*(t) = \varphi_0 + \varphi_1 t + \varphi_2 t^2 \quad t \in [t_1, t_3]$$

los coeficientes se calculan con el siguiente sistema de ecuaciones:

$$f_1 = \varphi_0 + \varphi_1 t_1 + \varphi_2 t_1^2 = 0 \quad 0 \leq t \leq t_1$$

$$f_2 = \varphi_0 + \varphi_1 t_2 + \varphi_2 t_2^2 = -1/2gt_2^2 \quad t_1 < t \leq t_2$$

$$f_3 = \varphi_0 + \varphi_1 t_3 + \varphi_2 t_3^2 = -1/2gt_3^2 \quad t_2 < t \leq t_3$$

Resolviendo el sistema anterior, la función  $R_3^*$  resulta de la siguiente forma:

$$R_3^*(t) = p_1(p_2 p_3),$$

donde;

$$p_1 = \frac{1}{2(t_1 - t_2)(t_1 - t_3)}$$

$$p_2 = g(t - t_1)$$

$$p_3 = -t_1 t_2 t_3 + t(-t_2 t_3 + t_1(t_2 + t_3))$$

Cabe mencionar, que los tiempos están definidos de la siguiente forma:  $t$  es el tiempo absoluto,  $t_1$  es el tiempo de retardo,  $t_2$  es el tiempo de intersección de la partícula TPL1, y  $t_3$  es el tiempo de intersección de la partícula TPL2.

Finalmente para encontrar la función de la partícula seguidora TPS, que es el algoritmo de recuperación de tiempos en términos de sus componentes, se tiene:

$$\text{TPS} = \text{TPS}_x + \text{TPS}_y, \quad (2.16c)$$

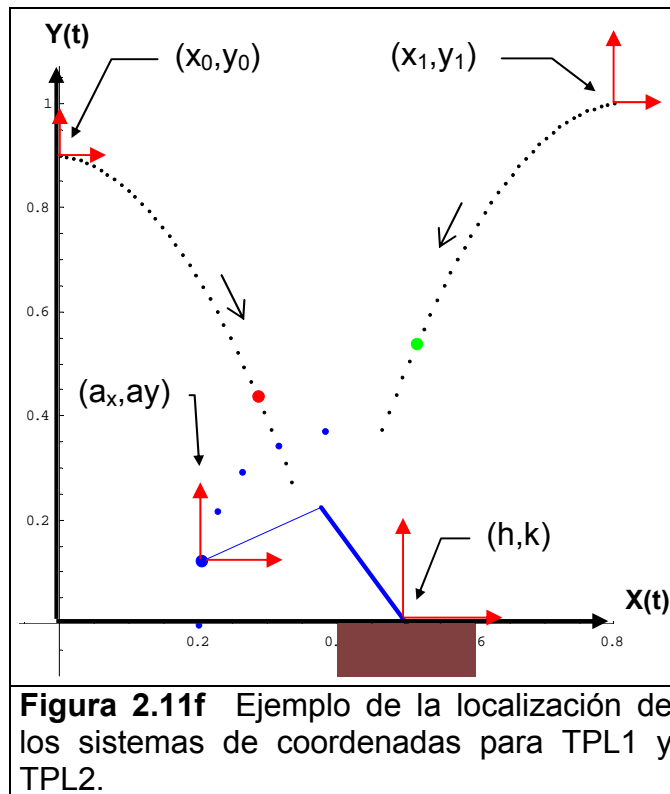
donde,

$$\text{TPS}_x = P^*_x + R^*_1$$

$$\text{TPS}_y = P^*_y + R^*_2 + R^*_3$$

$\text{TPS}_x = ax, \quad 0 \leq t \leq t_1$ $\text{TPS}_x = p_1(ax \cdot p_2 - (t - t_1)p_3 + t \cdot p_4 - p_7), \quad t_1 < t \leq t_3$
<p>donde;</p> $p_1 = \frac{1}{(t_1 - t_2)(t_1 - t_3)(t_2 - t_3)}$ $p_2 = (t - t_2)(t - t_3)(t_2 - t_3)$ $p_3 = (-t_1 t_3 + t_3^2)x_0 + (t_1 t_2 - t_2^2)x_1$ $p_4 = (t_1 - t_3)x_0 - (t_1 - t_2)x_1$ $p_5 = t_2(t - t_3)(t_1 - t_3)v_0 \cos \theta_0$ $p_6 = (t - t_2)(-t_1 + t_2)t_3 v_1 \cos \theta_1$ $p_7 = (t - t_1)(p_5 + p_6)$

$\text{TPS}_y = ay, \quad 0 \leq t \leq t_1$ $\text{TPS}_y = P^*_y(t) + R^*_2(t) + R^*_3(t), \quad t_1 < t \leq t_3$	
<p>Donde;</p> $P^*_y(t) = p_1(ay \cdot p_2 - (t - t_1)p_3 + t \cdot p_4),$ <p>donde</p> $p_1 = \frac{1}{(t_1 - t_2)(t_1 - t_3)(t_2 - t_3)}$ $p_2 = (t - t_2)(t - t_3)(t_2 - t_3)$ $p_3 = (-t_1 t_3 + t_3^2)y_0 + (t_1 t_2 - t_2^2)y_1$ $p_4 = (t_1 - t_3)y_0 - (t_1 - t_2)y_1$	
$R^*_2(t) = p_1((t - t_1)(t_2(p_2 + p_3))),$ <p>donde;</p> $p_1 = -\frac{1}{(t_1 - t_2)(t_1 - t_3)(t_2 - t_3)}$ $p_2 = (t - t_3)(t_1 - t_3)v_0 \sin \theta_0$ $p_3 = (t - t_2)(-t_1 + t_2)t_3 v_1 \sin \theta_1$	$R^*_3(t) = p_1(p_2 p_3),$ <p>donde;</p> $p_1 = \frac{1}{2(t_1 - t_2)(t_1 - t_3)}$ $p_2 = g(t - t_1)$ $p_3 = -t_1 t_2 t_3 + t(-t_2 t_3 + t_1(t_2 + t_3))$



A continuación se presenta un ejemplo de simulación utilizando la ecuación (2.16c). En la figura 2.11f, se muestra una imagen de un evento de intercepción en donde se ha considerado ahora la posibilidad de cambiar: la posición del efector final del robot interceptor con las coordenadas  $(a_x, a_y)$ , medidas desde el sistema de coordenadas fijo  $(X(t), Y(t))$ . Además, se puede cambiar las posiciones del inicio de los disparos de las partículas libres TPL1 y TPL2 con  $(x_0, y_0)$  y  $(x_1, y_1)$ . Y la posición de la base del robot se podrá cambiar con las coordenadas  $(h, k)$ . En la figura 2.11f, los valores de las coordenadas son:  $(h=0.5, k=0.0)$ ,  $(a_x=0.2, a_y=0.1)$ ,  $(x_0=0.0, y_0=0.9)$  y  $(x_1=0.8, y_1=1.0)$ .

## II.11 ÁLGEBRA DE QUATERNIONES PARA OBTENCIÓN DEL MODELO CINEMÁTICO DEL ROBOT INTERCEPTOR.

El álgebra de Cuaterniones fue desarrollada por Sir William Hamilton en el año de 1843. Sin embargo, no fue hasta la década de los 60's, del siglo XX, cuando se empezaron a aplicar a la dinámica de cuerpos rígidos articulados. Es importante mencionar, que tal aplicación no se realizaba de una manera sistemática, puesto que el álgebra de Cuaterniones era solo una extensión formal no rigurosa del álgebra de los números complejos.

En esta sección se presenta un resumen del álgebra de Cuaterniones con el enfoque moderno del álgebra lineal [2.6], con el fin de utilizarla para modelar la cinemática del robot interceptor, ya que las ecuaciones se presentan en forma de algoritmos computacionales fácilmente programables.

### II.11.1 ÁLGEBRA DE QUATERNIONES

En esta parte se presenta en forma resumida las propiedades de los Cuaterniones, según [2.7].

Sea el conjunto  $\mathfrak{R}^4$ , sobre el cual se definen las operaciones siguientes:

i). Una operación aditiva,  $\oplus : \mathfrak{R}^4 \times \mathfrak{R}^4 \rightarrow \mathfrak{R}^4$ , definida como:

$$(a,b,c,d) \oplus (\alpha,\beta,\chi,\delta) = (a+\alpha, b+\beta, c+\chi, d+\delta) \\ \forall (a,b,c,d), (\alpha,\beta,\chi,\delta) \in \mathfrak{R}^4. \quad (2.17)$$

ii). Una operación multiplicativa,  $*$  :  $\mathfrak{R}^4 \times \mathfrak{R}^4 \rightarrow \mathfrak{R}^4$ , definida como:

$$(a,b,c,d) * (\alpha,\beta,\chi,\delta) = (a\alpha - b\beta - c\chi - d\delta, a\beta + b\alpha + c\delta + d\chi, \\ a\chi - b\delta + c\alpha + d\beta, a\delta + b\chi - c\beta + d\alpha) \quad (2.18)$$

$$\forall (a,b,c,d), (\alpha,\beta,\chi,\delta) \in \mathfrak{R}^4$$

Por medio de las cuales la pareja  $(\mathfrak{R}^4, \oplus)$  y  $(\mathfrak{R}^4, *)$ , forman un grupo aditivo conmutativo y un grupo multiplicativo no conmutativo respectivamente, según se muestra en [2.8]. Las propiedades algebraicas de este conjunto se resumen en los siguientes resultados:

**Teorema 2.1** La terna  $(\mathfrak{R}^4, \oplus, *)$ , es un campo no conmutativo.

Demostración: En efecto,

i) La operación  $*$  :  $\mathfrak{R}^4 \times \mathfrak{R}^4 \rightarrow \mathfrak{R}^4$  es asociativa, ya que

$$p*(q*s) = (p*q) *s \quad \forall p, q, s \in \mathfrak{R}^4$$

ii) El elemento  $\mathbf{1} = (1,0,0,0) \in \mathfrak{R}^4$ , es tal que;  $\mathbf{1}*p = p*\mathbf{1} = p$ ,  $\forall p \in \mathfrak{R}^4$ .  
Este es llamado el elemento nulo multiplicativo en  $\mathfrak{R}^4$ .

iii)  $\forall p \in \mathfrak{R}^4, p \neq (0,0,0,0)$ ; existe  $p^{-1} \in \mathfrak{R}^4$ , tal que  $p * p^{-1} = \mathbf{1}$ . El ejemplo  $p^{-1}$  es llamado el inverso de  $p$  para la multiplicación.

iv) La operación  $*$  :  $\mathfrak{R}^4 \times \mathfrak{R}^4 \rightarrow \mathfrak{R}^4$  no es conmutativa, esto es;

$$p*q \neq q*p$$

v) Las siguientes propiedades de distributividad se satisfacen:

$$a) (p \oplus q) * s = p * s \oplus q * s$$

$$b) p * (q \oplus s) = p * q \oplus p * s, \quad \forall p, q, s \in \mathfrak{R}^4$$

Se presenta ahora la operación  $\bullet : \mathfrak{R} \times \mathfrak{R}^4 \rightarrow \mathfrak{R}^4$ , definida por:

$$\alpha \bullet (a,b,c,d) = (\alpha a, \alpha b, \alpha c, \alpha d), \quad \forall (a,b,c,d) \in \mathfrak{R}^4 \text{ y } \alpha \in \mathfrak{R}.$$

La cual es una multiplicación escalar.

Se observa con esto que la terna  $(\mathfrak{R}^4, \oplus, \bullet)$ , es un espacio vectorial real. Además, se sabe que la transformación  $\langle \bullet, \bullet \rangle : \mathfrak{R}^4 \rightarrow \mathfrak{R}$  definida por:

$$\langle \mathbf{p}, \mathbf{q} \rangle = \sum_{i=0}^3 p_i q_i \in \mathfrak{R}$$

es un producto interno en  $\mathfrak{R}^4$ , y por lo tanto,  $\mathbf{Q} = (\mathfrak{R}^4, \oplus, *, \bullet, \langle \bullet, \bullet \rangle)$  es un espacio vectorial real, y la norma asociada a este producto interno es la siguiente:

$$\|\mathbf{p}\| = \langle \mathbf{p}, \mathbf{p} \rangle^{1/2} = (p_0^2 + p_1^2 + p_2^2 + p_3^2)^{1/2}$$

por lo cual  $\mathbf{Q} = (\mathfrak{R}^4, \oplus, *, \bullet, \|\bullet\|)$  es un espacio normado, al cual llamaremos espacio vectorial de Cuaterniones y a sus elementos Cuaterniones.

Se representan los siguientes isomorfismos también en [2.6] apropiados para las aplicaciones, entre el espacio vectorial de cuaterniones, el espacio vectorial real y el espacio  $\mathfrak{R}^3$ , esto sirve para expresar un quaternion como la suma de un quaternion real y un quaternion vectorial. Esto es;

$$\mathbf{Q}_r = \{ (a, 0, 0, 0) : a \in \mathfrak{R} \} \subset \mathbf{Q}$$

$$\mathbf{Q}_v = \{ (0, b, c, d) : b, c, d \in \mathfrak{R} \} \subset \mathbf{Q}$$

Entonces las transformaciones definidas por:

$$\text{Tr}(a, 0, 0, 0) = a \quad \forall (a, 0, 0, 0) \in \mathbf{Q}_r$$

$$\text{Tv}(0, b, c, d) = (b, c, d) \quad \forall (0, b, c, d) \in \mathbf{Q}_v$$

Son isomórficas, por lo tanto, si  $\mathbf{p} = (p_0, p_1, p_2, p_3) \in \mathbf{Q}$ . El producto  $\mathbf{s} = \mathbf{p} * \mathbf{q}$  tiene otra representación:

$$\mathbf{s} = \mathbf{s}_r + \mathbf{s}_v \text{ y}$$

$$\mathbf{s}_r = \text{Tr}^{-1}[q_0 p_0 - \langle \mathbf{p}, \mathbf{q} \rangle]$$

$$\mathbf{s}_v = \text{Tv}^{-1}[ p_0 \mathbf{q} + q_0 \mathbf{p} + \mathbf{p} \times \mathbf{q} ]$$

donde  $\langle \bullet, \bullet \rangle : \mathfrak{R}^3 \times \mathfrak{R}^3 \rightarrow \mathfrak{R}$ , y  $\times : \mathfrak{R}^3 \times \mathfrak{R}^3 \rightarrow \mathfrak{R}^3$  son el producto interno y vectorial usual en  $\mathfrak{R}^3$ .

De manera similar, el álgebra de los números complejos se puede definir como un quaternion conjugado como:  $\bar{\mathbf{p}} = (p_0, -p_1, -p_2, -p_3)$ . La definición anterior permite mostrar el siguiente teorema con algunas propiedades útiles para el caso de estudio.

**Teorema 2.2** Dados  $\mathbf{p} = (p_0, p_1, p_2, p_3)$  y  $\mathbf{q} = (q_0, q_1, q_2, q_3) \in \mathbf{Q}$ , entonces:

- i)  $\overline{\mathbf{p} \oplus \mathbf{q}} = \overline{\mathbf{p}} \oplus \overline{\mathbf{q}}$
- ii)  $\overline{\mathbf{p} * \mathbf{q}} = \overline{\mathbf{p}} * \overline{\mathbf{q}}$
- iii)  $\mathbf{p} * \overline{\mathbf{p}} = \overline{\mathbf{p}} * \mathbf{p} \in \mathbb{Q}r$

en álgebra representada en esta sección, es de utilidad para las aplicaciones según se muestra en el siguiente párrafo:

### II.11.2 REPRESENTACIÓN PARAMÉTRICA DE ROTACIONES DE UN CUERPO RÍGIDO

Se intenta unificar y por consiguiente, simplificar las diferentes formas utilizadas en la mecánica para representar paramétricamente las rotaciones finitas de cuerpos rígidos, de forma similar a [2.6]. Para este, se define una transformación lineal  $\rho(\mathbf{p}, \bullet): \mathbf{Q} \rightarrow \mathbf{Q}$ ,  $\mathbf{p} \in \mathbf{Q}$  está fijo, que preserva el producto interno, la norma y el ángulo en la forma siguiente:

$$\rho(\mathbf{p}, \mathbf{q}) = \mathbf{p} * \mathbf{q} * \mathbf{p}^{-1} = \frac{1}{|\mathbf{p}|^2} \bullet (\mathbf{p} * \mathbf{q} * \overline{\mathbf{p}}) \quad \forall \mathbf{q} \in \mathbf{Q} \quad (2.7)$$

el siguiente teorema estudia las propiedades de la transformación definida en (2.6).

**Teorema 2.3** La transformación  $\rho(\mathbf{p}, \bullet): \mathbf{Q} \rightarrow \mathbf{Q}$  es lineal, ortogonal y  $\rho(\mathbf{p}, \bullet): \mathbf{Q}_v \rightarrow \mathbf{Q}_v$ , para todo  $\mathbf{q} \in \mathbf{Q}_v$ .

**Teorema 2.4** sea  $\theta \in [0, n\pi]$ ,  $n \in \mathbb{R}^+$ ,  $\mathbf{w} \in \mathbf{Q}_v$ ,  $|\mathbf{w}| = 1$ ; entonces, el quaternion  $\mathbf{p} \in \mathbf{Q}$  con  $p_0 \in \mathbb{R} - \{0\}$  y  $p_v = \pm \text{Sen } \theta \mathbf{w}$ , es tal que  $\rho(\mathbf{p}, \mathbf{w}) = \mathbf{w}$ . por otra parte,  $p_0$  satisface la ecuación:

$$4 p_0^4 - 4 p_0^2 |\mathbf{p}|^2 \text{Cos } \theta - |\mathbf{p}|^4 \text{Sen}^2 \theta = 0$$

por el teorema 2.4 [2.6], el quaternion  $\mathbf{p}$  representa una rotación de un ángulo  $\theta \in [0, n\pi]$ , con eje de giro  $\mathbf{w} \in \mathbf{Q}_v$ , esto es;

$$p_0 = \pm |\mathbf{p}| \text{Cos } (\theta/2)$$

$$p_v = \pm |\mathbf{p}| \text{Sen } (\theta/2) \mathbf{w}$$

en general la norma de  $\mathbf{p}$ , puede ser arbitraria. En este trabajo se utilizan Quaterniones unitarios, esto es;  $|\mathbf{p}| = 1$  en este caso;

$$p_0 = \pm \text{Cos } (\theta/2) \quad (2.8)$$

$$p_v = \pm \text{Sen } (\theta/2) \mathbf{w} \quad (2.9)$$

Los componentes del quaternion  $\mathbf{p} \in \mathbb{R}^3$  expresados por las ecuaciones 2.8 y 2.9 son conocidos como los parámetros de Euler.

### II.11.3 ISOMORFISMO DE LOS VECTORES DE $\mathfrak{R}^3$ AL ESPACIO VECTORIAL $\mathbf{Q}$ .

De las propiedades de los cuaterniones existe una transformación lineal  $t_v : Q_v \rightarrow \mathfrak{R}^3$ , entre el subespacio de los Cuaterniones vectoriales  $Q_v$  y el espacio vectorial  $\mathfrak{R}^3$ , tal que:

$$T_v(0,b,c,d) = (b,c,d) \quad \forall (0,b,c,d) \in \mathbf{Q}_v$$

Dado que esta transformación es un isomorfo, existe una transformación inversa:

$$t_v^{-1} : \mathfrak{R}^3 \rightarrow Q_v, \text{ tal que}$$

$$t_v^{-1}(b,c,d) = (0,a,b,c) \quad \forall (b,c,d) \in \mathfrak{R}^3$$

con lo anterior podemos considerar a cualquier vector dado en  $\mathfrak{R}^3$  como un elemento del espacio vectorial de los Cuaterniones. Así, al aplicarle el isomorfismo a la base canónica de  $\mathfrak{R}^3$ ,

$$\mathbf{B} = \{(1,0,0), (0,1,0), (0,0,1)\}, \text{ Obtenemos:}$$

$$e_2 = T_v^{-1}(1,0,0) = (0,1,0,0)$$

$$e_3 = T_v^{-1}(0,1,0) = (0,0,1,0)$$

$$e_4 = T_v^{-1}(0,0,1) = (0,0,0,1)$$

donde  $\{e_j\} \in \mathfrak{R}^4$ ,  $j=2,3,4$  son vectores de la base canónica de  $\mathfrak{R}^4$ . La cual se completa el vector  $e_1 = (1,0,0,0)$ . Es importante resaltar que el isomorfismo anterior permitirá trabajar los elementos  $\mathfrak{R}^3$ , utilizando las propiedades algebraicas del espacio de los Cuaterniones  $\mathbf{Q}$ .

### II.11.4 ROTACIÓN DE UN SISTEMA CARTESIANO DE REFERENCIA

En esta parte se presenta la forma de aplicar una rotación rígida con el álgebra de Cuaterniones, con el propósito de utilizarla después en rotaciones rígidas de vectores de  $\mathfrak{R}^3$ . La transformación  $\rho(\mathbf{p}, \bullet): \mathbf{Q} \rightarrow \mathbf{Q}$  definida por:

$$\rho(\mathbf{p}, \mathbf{q}) = \mathbf{p} * \mathbf{q} * \mathbf{p}^{-1} = \frac{1}{|\mathbf{p}|^2} \bullet (\mathbf{p} * \mathbf{q} * \bar{\mathbf{p}}) \quad \forall \mathbf{q} \in \mathbf{Q} \quad (2.10)$$

la cual es una rotación en el espacio vectorial de los Cuaterniones.

Donde:

$*$  :  $\mathbf{Q} \times \mathbf{Q} \rightarrow \mathbf{Q}$  Es la operación multiplicativa de los Cuaterniones.

$|\bullet|$ :  $\mathbf{Q} \rightarrow \mathfrak{R}^+$  es la norma en el espacio vectorial  $\mathbf{Q}$ .

$\bullet$  : es la multiplicación escalar y  $\bar{\mathbf{p}} \in \mathbf{Q}$  es el conjugado de  $\mathbf{p}$ .

Los componentes de  $\mathbf{p} \in \mathbf{Q}$  están relacionados con los parámetros físicos de la rotación, mediante las siguientes expresiones:



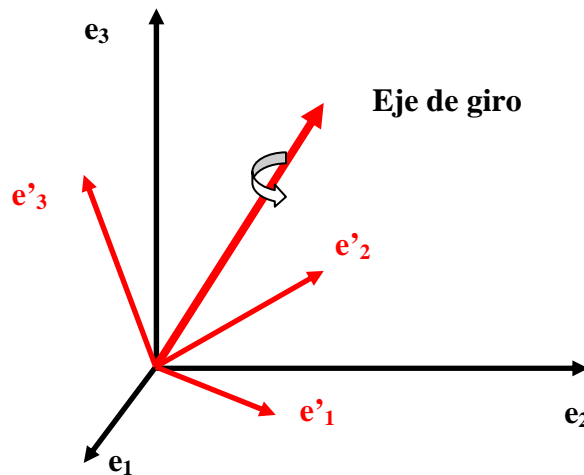
$$P_0 = \pm \text{Cos} (\theta/2), \quad p_v = \pm \text{Sen} (\theta/2)w$$

Donde  $\theta$  es el ángulo de rotación y  $w = (w_1, w_2, w_3)$  representa el eje de giro.

Si  $p \in \mathbf{Q}$  es un Quaternion unitario, entonces la rotación  $\rho(p, \bullet): \mathbf{Q} \rightarrow \mathbf{Q}$  se simplifica de la siguiente manera:

$$\rho(p, q) = p * q * \bar{p} \quad \forall q \in \mathbf{Q} \quad (2.11)$$

Esta operación es muy útil en la cinemática cuando se tiene que relacionar un sistema de referencia a partir de otro, relacionado mediante rotaciones la base que forma cada uno de estos sistemas. Esto se mostrará con el álgebra de Quaterniones y la base canónica  $\{e_j\}$  que se rotará para transformarse en la base  $(e'_j)$ ,  $j=1,2,3$ , como se muestra en la figura 2.11.



**Figura 2.11** Representación de la rotación del sistema de coordenadas  $\{e_j\}$ ,  $j = 1,2,3$  para convertirlo en  $\{e'_j\}$ .

Primero se transforma la base  $\{e_j\}$ ,  $j= 1,2,3$  en un conjunto  $\{e'_{j+1}\} \subset \mathbf{Q}$ ,  $j = 1,2,3$  bajo la siguiente relación

$$e^*_{j+1} = T_v^{-1} e_j$$

posteriormente se aplica a  $e^*_{j+1}$  la transformación  $\rho(p, \bullet): \mathbf{Q} \rightarrow \mathbf{Q}$  para obtener

$$a_{j+1} = \rho(p, e^*_{j+1}), j=1,2,3$$

la base  $a_{j+1} \subset \mathbf{Q}$ ,  $j = 1,2,3$  se completa al agregarle el vector  $a_1 = (1,0,0,0)$ . Ahora para obtener la base rotada  $\{e'_j\}$ ,  $j = 1,2,3$  basta aplicar a la base  $\{a_{j+1}\}$ ,  $j = 1,2,3$  la transformación  $T_v : \mathbf{Q}_v \rightarrow \mathfrak{R}^3$ ,  $e'_j = T_v a_{j+1}$ , obsérvese que los componentes de  $w$  se conocen respecto a la base  $\{e_j\}$ ,  $j = 1,2,3$ .

Esta es la forma de utilizar los isomorfismos de  $\mathfrak{R}^3$  y el subespacio de los cuaterniones vectoriales, para rotar un vector en  $\mathfrak{R}^3$  utilizando el álgebra de Cuaterniones.

### II.11.5 PROPIEDADES DE LA TRANSFORMACIÓN $\rho(\mathbf{p}, \bullet): \mathbf{Q} \rightarrow \mathbf{Q}$ [2.9,2.10]

El objetivo de esta parte es estudiar algunas de las propiedades de la transformación  $\rho(\mathbf{p}, \bullet)$ . La primera propiedad muestra la relación de la matriz asociada a la rotación  $\rho(\mathbf{p}, \bullet): \mathbf{Q} \rightarrow \mathbf{Q}$  denotada por  $M_{\rho(\mathbf{p}, \bullet): \mathbf{Q} \rightarrow \mathbf{Q}}$  con la operación multiplicativa de los Cuaterniones, la segunda y la tercera propiedad muestran las relaciones que se obtienen al derivar la norma de un Cuaternion, ésta es útil en la demostración de las relaciones cinemáticas. Las demás propiedades mostradas tienen que ver con la demostración de la aplicación de la regla de la cadena en la operación multiplicativa de los Cuaterniones.

Para lograr estos objetivos se define una transformación  $T_c : \mathbf{Q}_v \rightarrow \mathfrak{R}^4$ , tal que, dado un vector  $\mathfrak{R}^4$  bajo la transformación  $T_c$  es igual a:

$$T_c \mathbf{P} = \mathfrak{S}$$

Donde  $\mathfrak{S}$  es el vector coordenado de  $\mathbf{P}$  con respecto a la base canónica  $\mathbf{B}$  de  $\mathfrak{R}^4$ . Por lo tanto, la transformación inversa  $T_c^{-1}$  aplicada a un vector coordenado  $\mathfrak{S} \in \mathfrak{R}^4$  lo transforma a un vector en  $\mathfrak{R}^4$ .

$$T_c^{-1} \mathfrak{S} = \mathbf{P}$$

Con esta idea la matriz asociada a la rotación  $M_{\rho(\mathbf{p}, \bullet): \mathbf{Q} \rightarrow \mathbf{Q}}$  aplicada al vector coordenado de un Cuaternion  $\mathbf{q}(t)$  es igual al vector coordenado del producto

$\rho(t) * \mathbf{q}(t) * \bar{\rho}(t)$ , esto es:

$$M_{\rho(\mathbf{p}, \bullet): \mathbf{Q} \rightarrow \mathbf{Q}} \\ T_c [\mathbf{q}(t)] = T_c [\rho(t) * \mathbf{q}(t) * \bar{\rho}(t)],$$

Donde  $\mathbf{B} = \{e_j\}$ ,  $j = 1,2,3$  es la base canónica del espacio  $\mathfrak{R}^4$  y las funciones  $\mathbf{p}: \mathbf{J} \rightarrow \mathbf{Q}$ ,  $\mathbf{q}: \mathbf{J} \rightarrow \mathbf{Q}$ , con  $\mathbf{J} = [0, T]$  y  $\mathbf{Q}$  el espacio vectorial de Cuaterniones están definidas por:

$$\begin{aligned} \rho(t) &= ( \rho_0(t), \rho_1(t), \rho_2(t), \rho_3(t) ) \\ \mathbf{q}(t) &= ( q_0(t), q_1(t), q_2(t), q_3(t) ) \end{aligned}$$

Se presentan los siguientes teoremas que son de utilidad con el propósito de obtener el modelo cinemático del robot interceptor.

**Teorema 2.5** sea  $\rho(\mathbf{p}, \bullet): \mathbf{Q} \rightarrow \mathbf{Q}$  la rotación definida por (2.10) y  $M_{\rho(\mathbf{p}, \bullet): \mathbf{Q} \rightarrow \mathbf{Q}}$ , la matriz asociada, entonces

$$M_{\rho(\mathbf{p}, \bullet): \mathbf{Q} \rightarrow \mathbf{Q}} T_c [\mathbf{q}(t)] - T_c [\rho(t) * \mathbf{q}(t) * \bar{\rho}(t)] = 0$$

**Teorema 2.6** Dado  $p: J \rightarrow Q$ , entonces

$$2(p_0(t) \dot{p}_0(t) + p_1(t) \dot{p}_1(t) + p_2(t) \dot{p}_2(t) + p_3(t) \dot{p}_3(t)) = 0$$

**Teorema 2.7** Dado  $p: J \rightarrow Q$ , entonces

$2(\dot{p}_0(t)^2 + \dot{p}_1(t)^2 + \dot{p}_2(t)^2 + \dot{p}_3(t)^2 + p_0(t)\ddot{p}_0(t) + p_1(t)\ddot{p}_1(t) + p_2(t)\ddot{p}_2(t) + p_3(t)\ddot{p}_3(t)) = 0$   
 dado que la regla de la cadena se preserva en el espacio vectorial de las matrices  $M$  se demostrará que en la rotación  $\rho(p(t), q(t)) = p(t) * q(t) * \bar{p}(t)$  también se preserva.

**Teorema 2.8** sean  $p(t)$  y  $q(t) \in Q$ , entonces;

$$\frac{d}{dt} (M_{\rho(p, \cdot): Q \rightarrow Q} T_c [q(t)]) = T_c [\dot{p}(t) * q(t) * \bar{p}(t) + p(t) * \dot{q}(t) * \bar{p}(t) + p(t) * q(t) * \dot{\bar{p}}(t)]$$

es decir ;

$$\begin{aligned} & \dot{M}_{\rho(p, \cdot): Q \rightarrow Q} T_c [q(t)] + M_{\rho(p, \cdot): Q \rightarrow Q} T_c [\dot{q}(t)] - \\ & \dot{p}(t) * q(t) * \dot{\bar{p}}(t) + p(t) * \dot{q}(t) * \dot{\bar{p}}(t) + p(t) * q(t) * \dot{\bar{p}}(t) = 0 \end{aligned}$$

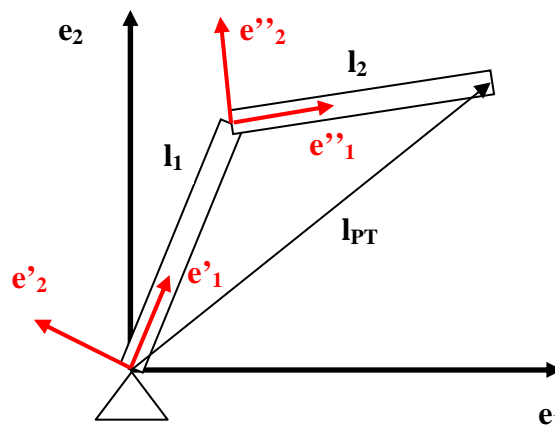
**Teorema 2.9** sea  $r = (0, r_1, r_2, r_3) \in Q_v$ , entonces:  $r = -\dot{\bar{r}}$

**Teorema 2.10** sea  $p: J \rightarrow Q$  y  $\bar{p}: J \rightarrow Q$ , su conjugado, entonces:

- a)  $\dot{p}(t) * \bar{p}(t) \in Q_v$
- b)  $\dot{p}(t) * \bar{p}(t) = -\dot{\bar{p}}(t) * p(t)$

Con los teoremas anteriores se puede modelar cadenas cinemáticas abiertas y cerradas.

**II.11.6 POSICIÓN, VELOCIDAD Y ACELERACIÓN DE DOS CUERPOS RÍGIDOS ACOPLADOS.**



**Figura 2.12** Sistema de referencia para la configuración 1.

Se consideran dos cuerpos acoplados por medio de dos juntas rotacionales y sujetos a un extremo de uno de ellos a una base fija como se muestra en la figura 2.12. Para iniciar la modelación se define un sistema de referencia global  $\{e_j\}$ ,  $j=1,2,3$  y los sistemas de referencia móviles sobre los eslabones  $\{e'_j\}$ ,  $j=1,2,3$  en el eslabón 1 y  $\{e''_j\}$ ,  $j=1,2,3$  para el eslabón 2 con el propósito de determinar el vector posición del extremo del efector final de robot respecto al sistema de referencia global.

La definición de los sistemas de referencia móviles se hizo de tal forma que los componentes de los vectores de posición de los eslabones cumplen con las siguientes expresiones:

$$\begin{aligned} l_{p1} &= l_1 \bullet e'_1 \\ l_{p2} &= l_2 \bullet e''_1 \end{aligned} \tag{2.12}$$

siendo  $l_1$  y  $l_2$  las longitudes de los eslabones. Aquí es importante destacar que el vector  $e'_1$  se definió paralelo al eslabón 1 de tal forma que la ecuación 2.12 es la que satisface dicha condición. De igual forma el vector de la base,  $e''_1$  se definió paralelo al eslabón 2.

El modelado consiste en determinar a través de los eslabones el vector  $l_{TP}$  del extremo del efector final.

$$l_{TP} = l_{p1} \oplus l_{p2} = l_1 \bullet e'_1 \oplus l_2 \bullet e''_1 \tag{2.13}$$

lo que a continuación se hará es transformar los elementos de las bases  $\{e'_j\}$ ,  $j=1,2,3$  y  $\{e''_j\}$ ,  $j=1,2,3$  a la base  $\{e_j\}$ ,  $j=1,2,3$  para obtener el  $l_{TP}$  en esta base.

Considerando que los sistemas de referencia locales (móviles), tienen relación sucesiva entre sí, con el sistema de referencia global. Entonces los elementos de estas bases estarán dados por:

$$\begin{aligned} e'_j &= Tv[\rho(\mathbf{p}, Tv^{-1}[e_j])] \\ e''_j &= Tv[\rho(\mathbf{p}, Tv^{-1}[e'_j])] = Tv[\rho(\mathbf{q}, \rho(\mathbf{p}, Tv^{-1}[e_j]))] \end{aligned}$$

donde  $\mathbf{p} = (p_0, p_1, p_2, p_3)$

$$p_0 = \cos \frac{\theta_1}{2} \quad \text{cantidad de rotación.}$$

$$p_v = \text{sen} \frac{\theta_1}{2} \mathbf{u} \quad \mathbf{u} = (0,0,1) \text{ es el eje de rotación de la base } e'_j. \text{ Eje } z.$$

$$\mathbf{q} = (q_0, q_1, q_2, q_3)$$

$$q_0 = \cos \frac{\theta_2}{2} \quad \text{cantidad de rotación.}$$

$$q_v = \text{sen} \frac{\theta_2}{2} \quad \mathbf{v}_j' \cdot \text{Tv}[\rho(\mathbf{p}, \text{Tv}^{-1}[\mathbf{e}_j])] = \text{sen} \frac{\theta_2}{2} \quad \mathbf{v}$$

$\mathbf{v} = (0,0,1)$  es el eje de rotación de la base  $\mathbf{e}'_j$ . Eje z.

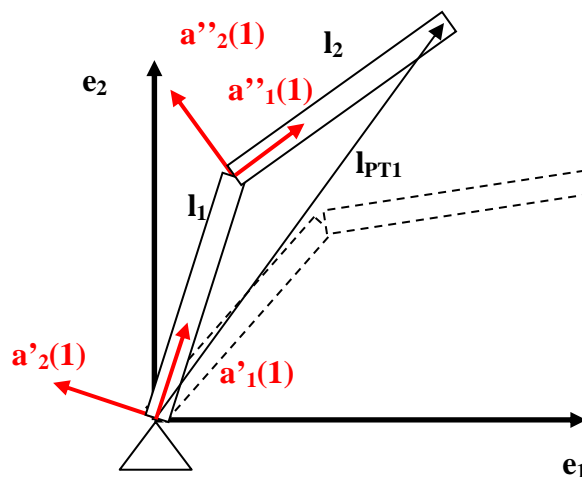
De la ecuación 2.12, se obtiene el vector de posición:

$$l_{PT} = l_{p1} + l_{p2} = l_1 \cdot \text{Tv}[\rho(\mathbf{p}, \text{Tv}^{-1}[\mathbf{e}_j])] \oplus l_2 \cdot \text{Tv}[\rho(\mathbf{q}, \rho(\mathbf{p}, \text{Tv}^{-1}[\mathbf{e}_j]))] \quad (2.14)$$

La ecuación 2.14 representa el modelo cinemático del robot interceptor de 2 gdl, en su configuración no deformada. Ahora, para aplicar movimiento a cada eslabón, se necesita el modelo deformado.

En cada configuración se identifica un vector de posición  $l_{TP}$ ,  $l_{TP1}$  respectivamente, se define desde la base al extremo final del eslabón  $l_2$ .

- a) Configuración 1. El modelo de esta configuración corresponde al no deformado ver figura 2.12.
- b) Configuración 2. Mostrada en la figura 2.13 corresponde a una rotación  $\psi_1$  del eslabón  $l_1$  respecto a su eje de rotación  $\mathbf{b}'$  definido en la base  $\{\mathbf{e}'_j\}$ ,  $j=1,2,3$ .



**Figura 2.13** Sistema de referencia para la configuración 2.

Por lo tanto, las bases  $\{e''_j\}$ ,  $\{e'_j\}$ ,  $j=1,2,3$ ; deben rotar  $\psi_1$  respectivamente para alcanzar la configuración deformada  $I_{TP1}$ , y están dadas por  $\{a'_j(1)\}$ ,  $\{a''_j(1)\}$ ,  $j=1,2,3$  como a continuación se muestra:

$$a'_j(1) = Tv[\rho(\mathbf{P}(t), Tv^{-1}[e'_j])] ]$$

$$a''_j(1) = Tv[\rho(\mathbf{P}(t), Tv^{-1}[e''_j])] ]$$

donde  $\mathbf{P}(t)$  esta constituido por:

$$\rho(t) = (p_0, p_1, p_2, p_3)$$

$$p_0 = \cos \frac{\psi_1}{2} \quad \text{cantidad de rotación}$$

$$p_v = \sin \frac{\psi_1}{2} \mathbf{b}_j' \bullet Tv[\rho(\mathbf{p}, Tv^{-1}[e_j])] ]$$

el eje de rotación  $\mathbf{b}'$  esta definido en el sistema  $e'_j$ , esto se indica de la siguiente manera  $\mathbf{b}' = \mathbf{b}_j' \bullet Tv[\rho(\mathbf{p}, Tv^{-1}[e_j])] ]$ .

Se observa que:

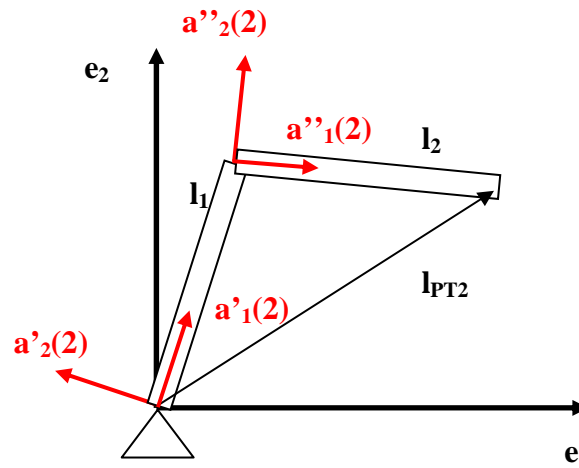
$$\mathbf{l}_1 = l_1 \bullet \mathbf{a}'_1(1)$$

$$\mathbf{l}_2 = l_2 \bullet \mathbf{a}''_1(1)$$

Por lo tanto, el vector de posición  $I_{TP1} \in \mathfrak{R}^3$  queda expresado por:

$$\begin{aligned} I_{TP1}(t) &= \mathbf{l}_1 \oplus \mathbf{l}_2 \\ &= Tv[\mathbf{l}_1 \bullet Tv[\rho(\mathbf{P}(t), \rho(\mathbf{p}, Tv^{-1}[e_1])])] \oplus \mathbf{l}_2 \bullet Tv[\rho(\mathbf{P}(t), \rho(\mathbf{q}, \rho(\mathbf{p}, Tv^{-1}[e_1])])] ] \quad (2.15) \end{aligned}$$

c) Configuración 3. mostrada en la figura 2.14, corresponde a una rotación de ángulo  $\psi_2$  del eslabón  $l_1$  respecto a su eje de rotación  $c''$  definido en la base  $\{e''_j\}$ ,  $j=1,2,3$



**Figura 2.14** Sistema de referencia para la configuración 3.

Por lo tanto, las bases  $\{a'_j(1)\}$ ,  $\{a''_j(1)\}$ ,  $j=1,2,3$ ; están dadas por  $\{a'_j(2)\}$ ,  $\{a''_j(2)\}$ ,  $j=1,2,3$  como a continuación se muestra.

$$a'_j(2) = a'_j(1)$$

$$a''_j(2) = Tv[\rho(\mathbf{Q}(t), Tv^{-1}[a''_j(1)])],$$

donde el quaternion  $\mathbf{Q}(t)$  está definido como:

$$\mathbf{Q}(t) = (Q_0, Q_1, Q_2, Q_3)$$

$$Q_0 = \cos \frac{\psi_2(t)}{2} \quad \text{cantidad de rotación.}$$

$$Q_v = \text{sen} \frac{\psi_2(t)}{2} \mathbf{c}_j \cdot Tv[\rho(\mathbf{P}(t), \rho(\mathbf{q}, \rho(\mathbf{p}, Tv^{-1}[e_1])))]$$

Donde el vector  $\mathbf{c}_j$  es el eje de rotación y está definido en el sistema y le ha afectado la rotación  $\mathbf{P}(t)$ .

Observando que:

$$l_1 = l_2 \bullet a'_1(2)$$

$$l_2 = l_2 \bullet a''_1(2)$$

Por lo tanto, el vector de posición  $l_{PT2} \in \mathbb{R}^3$  y la ecuación 2.12, queda expresado por:

$$l_{TP2}(t) = T_v [P(t) * T_v^{-1} [l_{P1}] * \bar{P}(t) + Q(t) * P(t) * T_v^{-1} [l_{P2}] * \overline{Q(t) * P(t)}] \quad (2.16)$$

La velocidad lineal de los eslabones está dada por la derivada del vector  $l_{TP2}(t)$  con respecto al tiempo:

$$\frac{d}{dt} l_{TP2}(t) = \frac{d}{dt} T_v [P(t) * T_v^{-1} [l_{P1}] * \bar{P}(t) + Q(t) * P(t) * T_v^{-1} [l_{P2}] * \overline{Q(t) * P(t)}] \quad (2.17)$$

la ecuación puede describirse de la siguiente forma[2.11,2.12,2.13]:

$$\dot{L}_{PT2}(t) = w_1(t) \times l_1 + w_2(t) \times l_2 + w_{R1}(t) \times l_2 \quad (2.18)$$

$$w_1(t) = Tv[2 \bullet \dot{p}(t) * \bar{p}(t)]$$

$$w_2(t) = Tv[2 \bullet \dot{Q}(t) * \bar{Q}(t)]$$

$$w_{R1}(t) = Tv[2 \bullet Q(t) * \dot{P}(t) * \bar{P}(t) * \bar{Q}(t)]$$

donde  $w_1(t)$  y  $w_2(t)$  son las velocidades angulares del primer y segundo eslabón, y  $w_{R1}(t)$  es la velocidad relativa del segundo cuerpo respecto al primero. La aceleración del sistema de cuerpos acoplados se obtiene al derivar respecto al tiempo la ecuación de velocidad dada en la ecuación (2.18).

$$\frac{d}{dt} \dot{L}_{PT2}(t) = \frac{d}{dt} w_1(t) \times l_1 + w_2(t) \times l_2 + w_{R1}(t) \times l_2$$

y nuevamente la ecuación puede describirse [2.11,2.12,2.13], como

$$\ddot{L}_{PT2}(t) = \alpha_1(t) \times l_1 + w_1(t) \times w_1(t) \times l_1 + \alpha_2(t) \times l_2 + w_2(t) \times w_2(t) \times l_2 + \alpha_{R1}(t) \times l_2 + w_{R1}(t) \times w_{R1}(t) \times l_2 \quad (2.19)$$

donde ;

$$w_1(t) = Tv[2 \bullet \dot{p}(t) * \bar{p}(t)] \quad \text{es la velocidad angular en } l_1$$

$$\alpha_1(t) = Tv[2 \bullet \ddot{p}(t) * \bar{p}(t)] \quad \text{es la aceleración angular en } l_1$$

$$w_2(t) = Tv[2 \bullet \dot{Q}(t) * \bar{Q}(t)] \quad \text{es la velocidad angular en } l_2$$

$$\alpha_2(t) = Tv[2 \bullet \ddot{Q}(t) * \bar{Q}(t)] \quad \text{es la aceleración angular en } l_2.$$

$$w_{R1}(t) = Tv[2 \bullet Q(t) * \dot{P}(t) * \bar{P}(t) * \bar{Q}(t)]$$

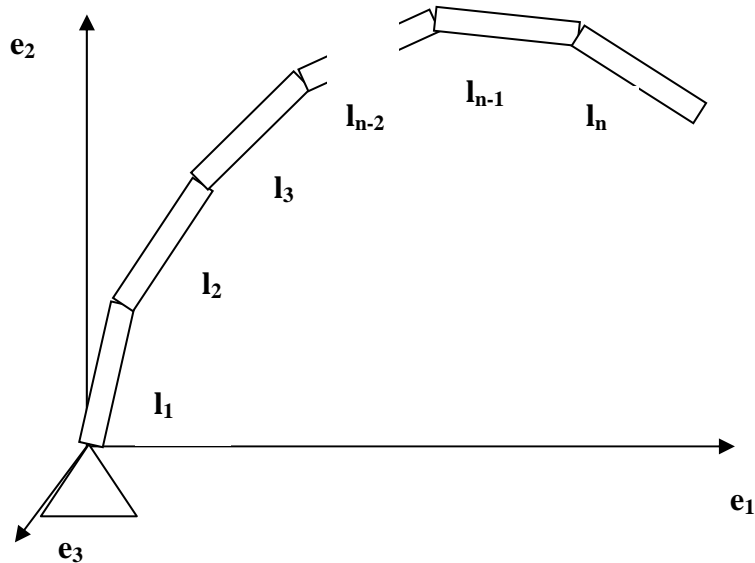
$$\alpha_{R1}(t) = Tv[2 \bullet Q(t) * \ddot{P}(t) * \bar{P}(t) * \bar{Q}(t)]$$

de esta manera se ha realizado una modelación general de dos cuerpos acoplados para obtener sus ecuaciones de posición, velocidad y aceleración en cada instante.



**II.12 ALGORITMO DE MODELADO DE N CUERPOS RÍGIDOS CON QUATERNIONES [2.13]**

En las secciones anteriores se desarrollaron las ecuaciones cinemáticas para dos cuerpos acoplados. Ahora se muestra un algoritmo para obtener estas ecuaciones para N cuerpos acoplados, se muestra en la figura 2.15. Haciendo un resumen de las secciones anteriores, tenemos las ecuaciones de posición, velocidad y aceleración con Quaterniones  $P_1, P_2$  con el propósito de sistematizar la obtención de las mismas para N-cuerpos.



**Figura 2.15** Sistema de N cuerpos acoplados

<b>Tabla 2.1</b> Metodología para la obtención del modelo cinemático de robot con n gdl.			
<b>No. cuerpos</b>	<b>Posición</b>	<b>Velocidad</b>	<b>Aceleración</b>
1	$Tv[P_1 * l_1 * \bar{P}_1]$	$Tv[2\dot{P}_1 * \bar{P}_1 * l_1] = w_{11} x l_1$	$Tv[2\ddot{P}_1 * \bar{P}_1 * l_1 + 4 \dot{P}_1 * \bar{P}_1 * \dot{P}_1 * \bar{P}_1 * l_1] = w_{11} x w_{11} x l_1$
2	$Tv[P_1 * l_1 * \bar{P}_1 + P_2 * P_1 * l_1 * \bar{P}_2 * \bar{P}_1]$	$Tv[2\dot{P}_1 * \bar{P}_1 * l_1 + 2\dot{P}_2 * \bar{P}_2 * l_2 + 2P_2 * \dot{P}_1 * \bar{P}_1 * \bar{P}_2 * l_2] = w_{11} x l_1 + w_{22} x l_2 + w_{12} x l_2$	$= \dot{w}_{11} x l_1 + w_{11} x w_{11} x l_1 + \dot{w}_{22} x l_2 + w_{11} x w_{11} x l_2 + \dot{w}_{12} x l_2 + w_{12} x w_{12} x l_2$

Es claro que para encontrar las ecuaciones de posición de sistemas con más de dos cuerpos se tiene que anexar las ecuaciones que cada cuerpo define con relación a los demás. Por lo tanto, a continuación se calculan solo las ecuaciones que cada N-ésimo elemento origina, considerando lo antes mencionado, ver tabla 2.1.

En un sistema de tres cuerpos acoplados el tercer elemento está dado por:

$$L_3 = Tv[P_3 * P_2 * P_1 * T_v^{-1}[l_3] * P_1 * P_1 * P_3]$$

Derivando y simplificando con el procedimiento en [2.8] con las propiedades de los Cuaterniones, se tiene que la velocidad que el cuerpo adquiere es:

$$\begin{aligned} \dot{L}_3 &= Tv[(\dot{P}_3 * P_2 * P_1 + P_3 * \dot{P}_2 * P_1 + P_3 * P_2 * \dot{P}_1) * \bar{P}_1 * \bar{P}_2 * \bar{P}_3] * T_v^{-1}[l_3] + \\ &\quad T_v^{-1}[l_3] * P_3 * P_2 * P_1 * (\dot{\bar{P}}_1 * P_2 + \bar{P}_3 * P_1 * \dot{\bar{P}}_2 * \bar{P}_3 + \bar{P}_1 * \bar{P}_2 + \dot{P}_3) \\ &= Tv[(2 \bullet \dot{P}_3 * \bar{P}_3 * T_v^{-1}[l_3] + 2 \bullet P_3 * \dot{P}_2 * \bar{P}_2 + \bar{P}_3 * T_v^{-1}[l_3] \\ &\quad + 2 \bullet P_3 * P_2 * \dot{P}_1 * \bar{P}_1 * \bar{P}_2 * \bar{P}_3 * T_v^{-1}[l_3]) \\ &= w_{33} \times l_3 + w_{23} \times l_3 + w_{13} \times l_3 \end{aligned}$$

La aceleración de este cuerpo es:

$$\ddot{L}_3 = \dot{w}_{33} \times l_3 + w_{33} + w_{33} \times l_3 + \dot{w}_{23} \times l_3 + w_{23} \times w_{23} \times l_3 + \dot{w}_{13} \times l_3 + w_{13} \times l_3$$

Entonces de esta generalización se puede inducir lo siguiente:

La ecuación de posición, velocidad y aceleración para **N** cuerpos acoplados son:

$$\text{Posición} = \sum_{m=1}^N L_m$$

$$\text{Velocidad} = \sum_{m=1}^N \dot{L}_m = \sum_{m=1}^N \sum_{i=1}^m w_{im} \times l_m$$

$$\text{Aceleración} = \sum_{m=1}^N \ddot{L}_m = \sum_{m=1}^N \sum_{i=1}^m (\alpha_{im} \times l_m + w_{im} \times w_{im} \times l_m), \text{ donde:}$$

$$L_m = P_m * P_{m-1} * \dots * P_3 * P_2 * P_1 * I_N * \overline{P_m * P_{m-1} * \dots * P_3 * P_2 * P_1}$$

$$L_m = \prod_{n=1}^m P_{m+1-n} * I_N * \prod_{n=1}^m \bar{P}_{m+1-n}$$

$$\dot{L}_m = \sum_{i=1}^m w_{im} \times l_m$$

$$\ddot{L}_m = \sum (\alpha_{im} \times l_m + w_{im} \times w_{im} \times l_m)$$

$$w_{im} = \left\{ \begin{array}{l} i = m \\ i \neq m \end{array} \right. \left. \begin{array}{l} Tv[2 \bullet \dot{P}_i * \bar{P}_i] = \dot{\theta}_i u_i \\ Tv[2 \bullet P_m * P_{m-1} * \dots * P_{i+1} * P_i * P_i * \bar{P}_m * \bar{P}_{m-1} * \dots * \bar{P}_{i+1}] \\ = Tv[\rho(\{P_m, P_{m-1}, \dots, P_{i+1}\}, Tv^{-1}[\theta_i u_i])] \end{array} \right\}$$

Nota:  $U_i$  es el eje de rotación.

$$\alpha_{im} = \left\{ \begin{array}{l} i = m \\ m - i = 1 \\ m - i > 1 \end{array} \right. \left. \begin{array}{l} Tv[2 \bullet \ddot{P}_m * \bar{P}_m] = \ddot{\theta}_i u_i \\ Tv[\rho(\{P_m, P_{m-1}, \dots, P_{i+1}\}, T^{v-1}[\alpha_{ii}]) + w_{mm} x w_{im}] \\ \rho(\{P_m, P_{m-1}, \dots, P_{i+1}\}, T^{v-1}[\alpha_{ii}]) + w_{mm} x w_{im} + \\ \left[ \sum \rho(\{P_m, P_{m-1}, \dots, P_{a+1}\}, Tv^{-1}[w_{aa}]) x Tv[\rho(\{P_a, P_{a-1}, \dots, P_{a+1}\}, Tv^{-1}[w_{ii}])] \right] \end{array} \right\} \quad (2.20)$$

La ecuación (2.20) muestra el algoritmo que sistematiza el cálculo de las ecuaciones de posición, velocidad y aceleración de N cuerpos acoplados.

## II.13 SUMARIO

La obtención de los algoritmos de recuperación de tiempos en términos de funciones seguidoras, apoyada en la metodología planteada, fue satisfactoria. Se resolvió el problema de la intercepción de una partícula cuyo evento inició desde el origen de los sistemas coordenados que coincidían. Luego, se aplicó para la intercepción de dos partículas con cambio de base.

También se presentaron las ecuaciones que involucran el desplazamiento del sistema de coordenadas  $X^*(t)$  sobre el eje  $X(t)$  y  $Y^*(t)$  sobre el eje  $Y(t)$ . Aunque la simulación sólo presentó el eje  $X^*(t)$  trasladado  $ax$  unidades, se incluyó el desarrollo para el desplazamiento  $ay$  unidades sobre el eje  $Y(t)$ .

Una aportación importante, consiste en el hecho de interceptar dos partículas con una sola trayectoria, utilizando la misma metodología pero con ciertas modificaciones. A esa tarea se le denominó *algoritmo de intercepción múltiple de trayectorias conocidas*, incluyendo también el análisis cinemático de las partículas TPL1, TPL2 y la partícula seguidora TPS.

La importancia del estudio de intercepción de trayectorias radica en que pueden ser aplicadas en la intercepción de piezas que viajan sobre las bandas de los Sistemas de Manufactura Flexibles (FMS), implementando la función de intercepción TPS en la cinemática inversa del robot interceptor y utilizando retroalimentación visual para obtener los parámetros cinemáticos de las trayectorias de las piezas. Esto implica recuperar los tiempos de espera de los vagones que contienen las piezas. Sin embargo, cabe hacer notar que la intercepción se logra con colisión (impacto) de la partícula seguidora con la partícula libre 1 TPL1, no obstante, la partícula libre 2 TPL2 recibe un impacto menor al de la partícula TPL1.

Por otra parte, se presentó el algoritmo para modelar robot interceptores de  $N$  grados de libertad utilizando el álgebra de Cuaterniones. La ventaja de utilizar el álgebra de cuaterniones radica en que los modelos cinemáticos obtenidos se presentan en forma de algoritmos computacionales, además éstos pueden ser simplificados como si fueran ecuaciones algebraicas. Se utilizan parámetros y los ángulos se definen dentro de los parámetros de tal forma que quedan implícitos en las ecuaciones.

Los resultados y desarrollos de este capítulo permitirán modelar el robot industrial Mitsubishi para implementarlo en la parte experimental de este trabajo, ya que este robot tiene cinco grados de libertad y su manipulación con matrices se vuelve complejo. Sin embargo, se presenta también el cálculo matricial como otra alternativa para modelar robot manipuladores.

## II.14 REFERENCIAS

- [2.1] D. Hujic, E. Croft, G. Zak, R.G. Fenton, J.K. Milis, B. Benhabib, THE ROBOTIC INTERCEPTION OF MOVING OBJECTS IN INDUSTRIAL SETTINGS: STRATEGY DEVELOPMENT AND EXPERIMENT. January, 1995.
- [2.2] Russell L Anderson, A Robot Ping Pong Player: Experiments in Real-Time Intelligent Control. Cambridge, MA: MIT Press, 1988.
- [2.3] Mehran Mehrandezh, Naftafi M. Sela, Robert G. Fenton y Beno Benhabib, "Robotic Interception of Moving Objects Using an Augmented Ideal Proportional Navigation Guidance Technique", IEEE Transactions on Systems, Man and Cybernetics --Part a: Systems and Humans, vol. 30, No. 3, p:238-250, Mayo 2000.
- [2.4] Won Hong, "Robotic Catching and Manipulation Using Active Visión", Bachelor of Science, University of California at Berkeley, Certificate by Jean-Jacques. E. Slotine, p: 65, 1992.
- [2.5] Roberto Rodríguez Rodríguez, "Interceptación de Trayectorias Tridimensionales". tesis de maestría, DEPMI-UNAM, -asesor: Dr. Luis Reyes Ávila, p: 31-36, 1996
- [2.6] Reyes A. L., Quaternions: "Une representation parametrique systematique des rotations finies", Partie 1:Le cadre Theorique. Partie 2: Quelques Aplications. Rapports de Recherche, INRIA. Octubre 1990.
- [2.7] Mauricio C. M, "Dinámica de cuerpos rígidos con Quaterniones: aplicación a los mecanismos". Escuela de graduados e investigación, Universidad Anáhuac del Sur, Reporte No. EGI-001/TDI/MM-0498.
- [2.8] Mario M. M, "Comparación entre las soluciones numérica y simbólica de la cinemática inversa de un robot de dos grados de libertad", Escuela de graduados e investigación, Universidad Anáhuac del Sur, Reporte No. EGI-005/TDI/AR-O4-0998.
- [2.9] Kupiers, J.B., "Quaternions and Rotation Sequences", Princeton University Press, Princeton, New Jersey. 1999.
- [2.10] Wheeter, M.D., "Iterative Estimation of Rotation and Translation using the Quaternion" ,.school of Computer Science, Carnegie Mellon University Pittsburgh, PA, 1995
- [2.11] Nóyes, P., "From Bit-String to Quaternions", Stanford Linear Accelerator Center, Stanford University, 1991.
- [2.12] Joy, K., "Quaternions", Visuatization and Graphic 'Research laboratory, Department of Computer Science. University of California, Davis, 1994.
- [2.13] Paul, G., Ikeuchi, K, "Representing the Motion of Objects using Dual Quatemions and its Applications", Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, 1997.
- [2.14] M.W. Fossier, "The Development of radar Homing Missiles", Journal of Guidace, control and dynamics, Vol.7, No.5, p:641-651, 1984.
- [2.15] W. Haeussermann, "Developments in the field of automatic Guidance and control of rockets", Journal of Guidance, control and dynamics, Vol. 4, No.3, p:255-239, 1981.
- [2.16] Paul Zarchan, "Tactical and strategic missile guidance", America institute of Aeronautics and Astronautics, loc. Progress in astronautics and aeronautics, 4<sup>ta</sup> edición, Vol. 199, ISBN 1-56347-497-2.
- [2.17] Lawrence D. Stone, Carl A. Barlow, Bayesian Multiple Target Tracking", Library of Congress Cataloging-in-Publication Data, 1999. ISBN 1-58053024-9

## CAPÍTULO III

# ANÁLISIS NUMÉRICO PRELIMINAR

### 3.1 GENERALIDADES

En este capítulo, se desarrolla un simulador diseñado especialmente para la realización de eventos del fenómeno de intercepción múltiple. Se hacen las pruebas con valores numéricos diferentes con el fin de validar los algoritmos de recuperación de tiempos desarrollados en el capítulo anterior, y para estudiar y analizar las posibles configuraciones mecánicas y cinemáticas del robot interceptor de dos grados de libertad. Se implementan los algoritmos en el robot interceptor a nivel de simulación computacional y se obtienen resultados numéricos.

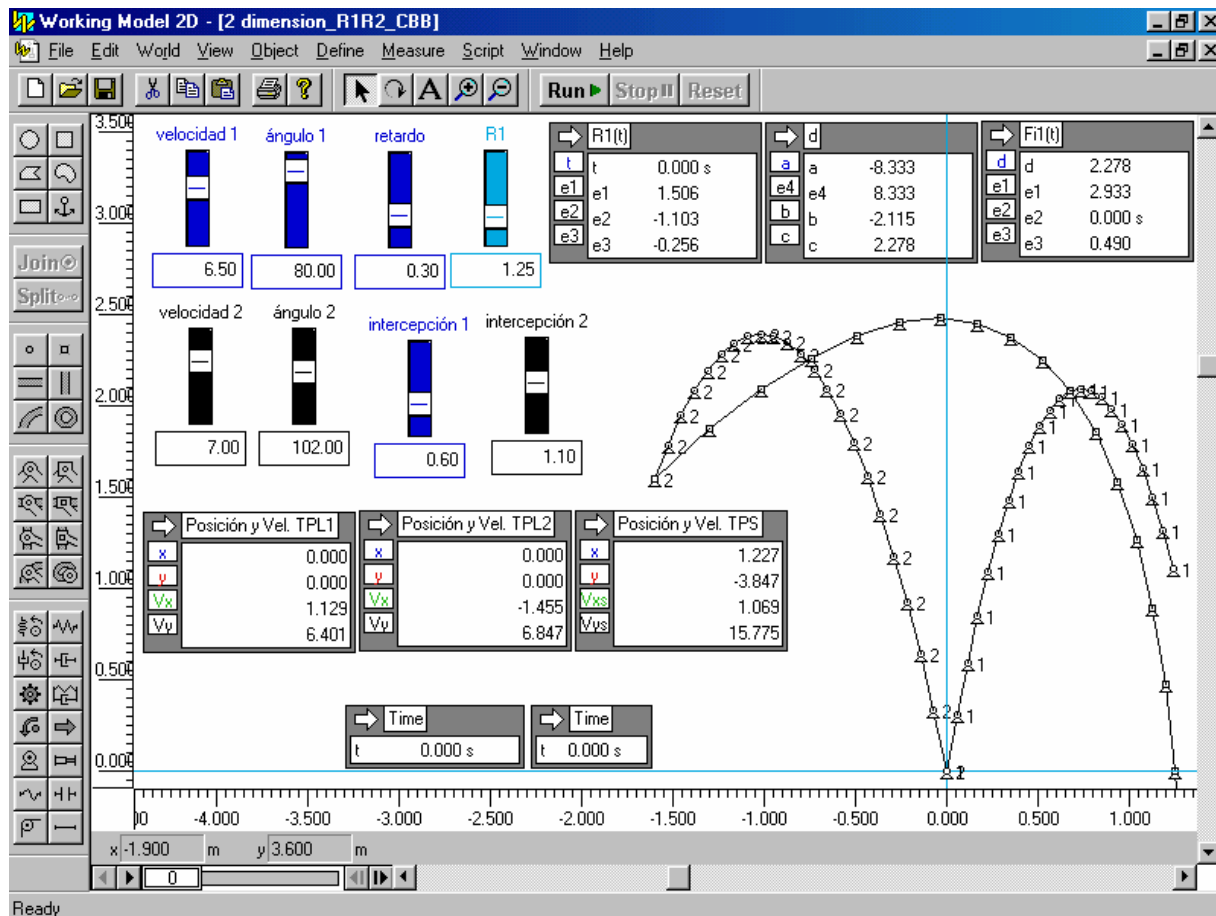
Se analizan los casos I y II presentados en el capítulo II para la intercepción de una partícula libre, después se realizan varias simulaciones con valores diferentes integrando un robot manipulador de dos gdl. Se utiliza el cambio de base con respecto al sistema de coordenadas fijo. Posteriormente, se estudia el primer punto a alcanzar del efector final en el fenómeno de intercepción múltiple para dos partículas. Luego se plantea al final, la propuesta de un esquema de control PID para controlar el movimiento del robot interceptor sobre la trayectoria.

Los desplazamientos angulares de los eslabones del robot interceptor se obtienen de los ángulos generados a partir de la cinemática inversa y aplicando interpolación cúbica. También, se obtienen las funciones de posición parabólica, velocidad y aceleración en función del tiempo de cada uno de los eslabones del robot interceptor de dos gdl.

La idea principal de este capítulo es la de acoplar los algoritmos de recuperación de tiempos al movimiento del robot interceptor de dos grados de libertad y posteriormente a uno de cinco grados de libertad. Lo anterior, con el fin de implementar la parte experimental y teniendo de base los resultados de la simulación.

### III.2 SIMULADOR DE FENÓMENO DE INTERCEPCIÓN MÚLTIPLE

En la figura 3.1 se muestra la simulación del evento de intercepción múltiple de dos partículas, el simulador fue desarrollado en el software comercial Working Model<sup>®</sup> 2D.



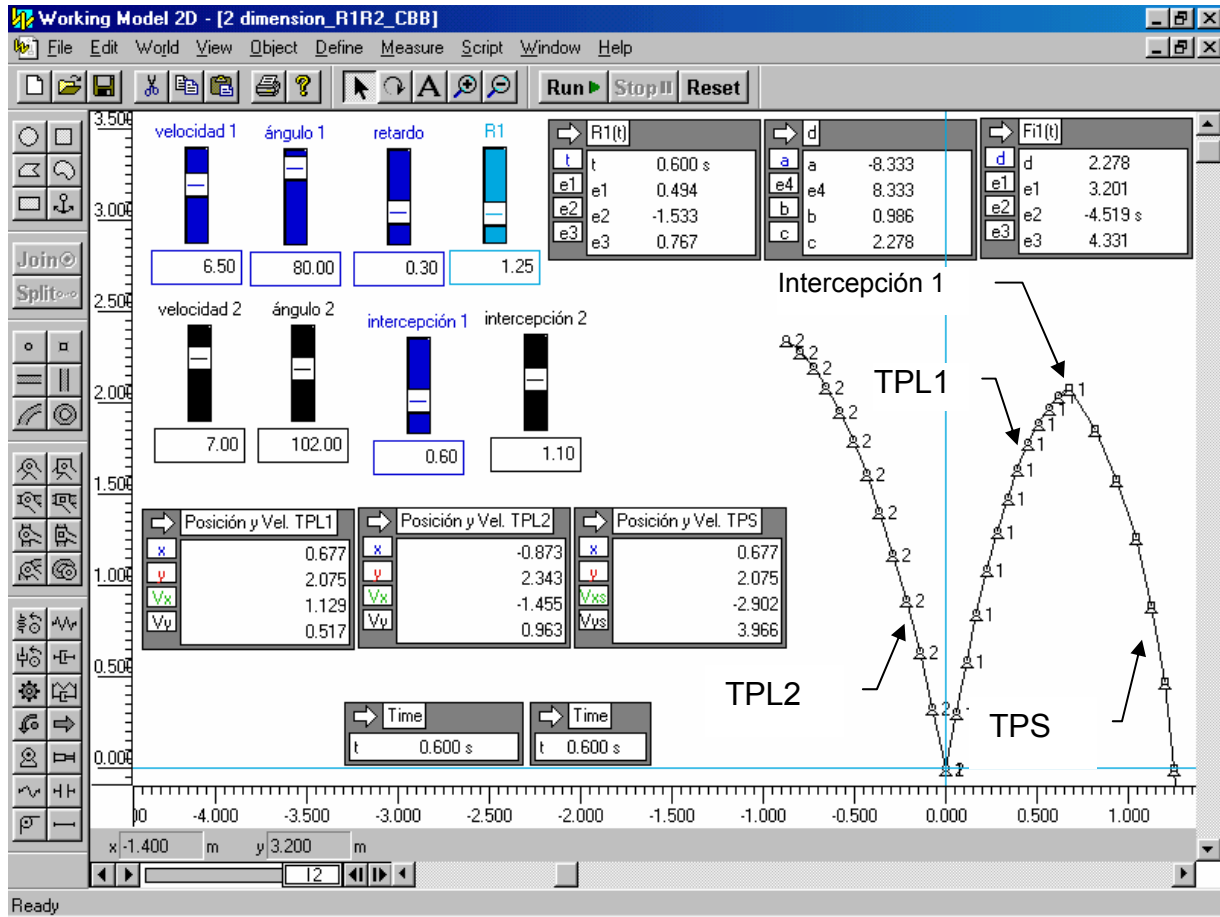
**Figura 3.1** Simulador para realización de eventos de intercepción múltiple.

El simulador se desarrolló con la finalidad de estudiar y analizar el comportamiento de la intercepción múltiple. En el simulador se puede variar las velocidades iniciales de las partículas TPL1 Y TPL2 en forma independiente, los ángulos de disparo para cada partícula, el tiempo de retardo de todo el evento y los tiempos de intercepción de cada partícula por separado. El simulador registra las variables de posición, velocidad y aceleración en función del tiempo de la partícula seguidora y de las partículas libres, además genera sus gráficas en cada evento realizado. La información de la partícula seguidora TPS se implementa, de tal forma, que el efector final del robot la siga para lograr la intercepción de las partículas. Luego, uno de los objetivos del simulador es que permite visualizar cuales son las posiciones adecuadas del efector final del robot, de su base, de la posición inicial de disparo de las partículas libres y sobre todo de la configuración del área de trabajo del robot interceptor en función de la longitud de sus eslabones.

Cabe mencionar, que para resolver el sistema de ecuaciones se implementó el método

numérico de Runge-Kutta de cuarto orden con un periodo de integración de 0.05s.

A continuación se presentan las simulaciones tentativas para analizar el comportamiento del fenómeno de intercepción. De esta manera se tendrá una visión más clara y amplia para el diseño del robot interceptor y su ubicación en el plano de trabajo.



**Figura 3.1a** Primer partícula interceptada en el tiempo 0.6 s.

En la figura 3.1a se muestra como la trayectoria de la partícula seguidora TPS alcanza e intercepta a la partícula libre 1, TPL1 en el instante de tiempo  $t_2 = 0.6$  s. Es importante mencionar que existe un tiempo de retardo para la partícula seguidora TPS de  $t_1 = 0.3$  s, antes de que inicie su movimiento.

En el simulador también se registran las posiciones en cada instante de tiempo. Es decir, en cada periodo de integración que en este caso es de 0.05 s. además, de que se lleva un registro de las velocidades de las partículas libres y la partícula seguidora.



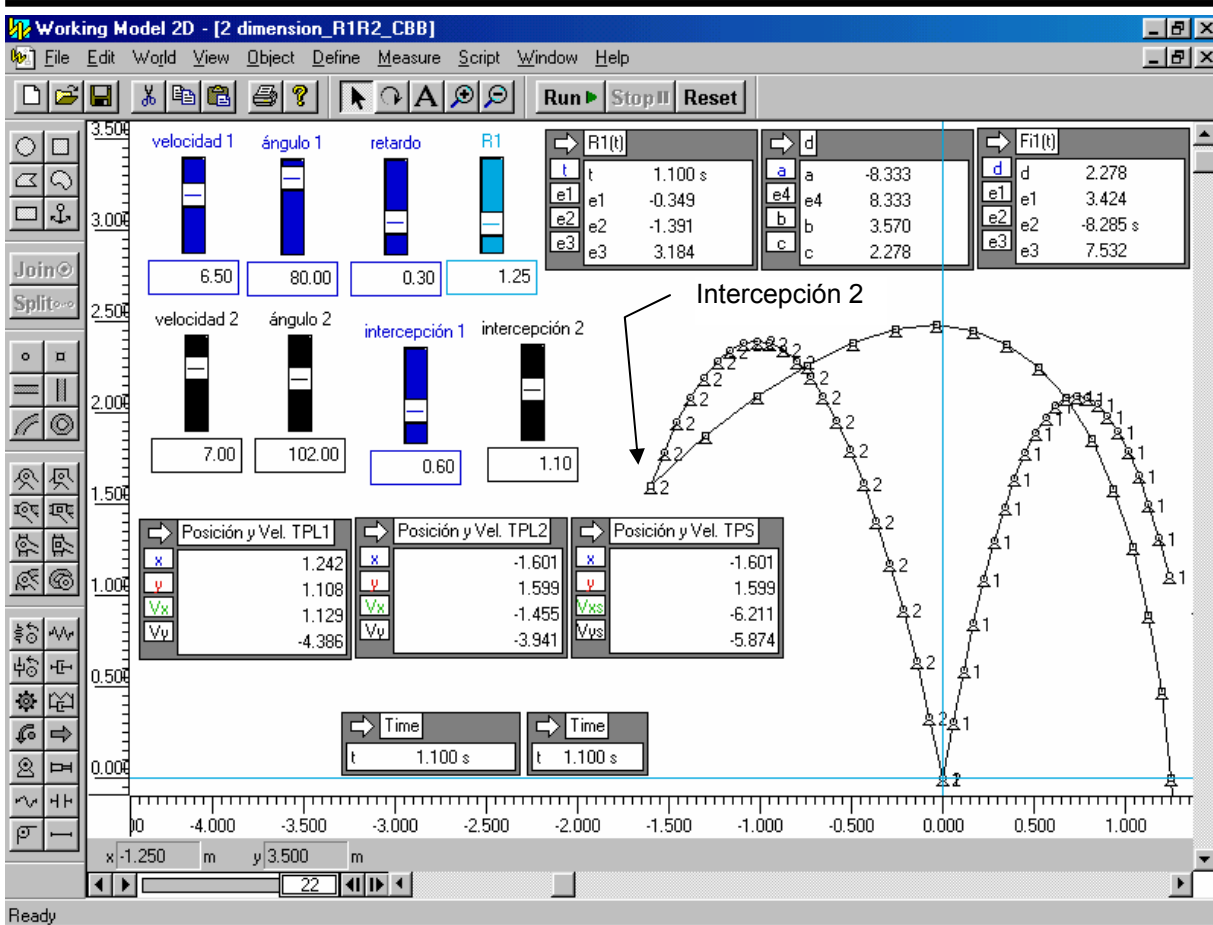


Figura 3.1b Primer partícula interceptada en el tiempo 1.1 s.

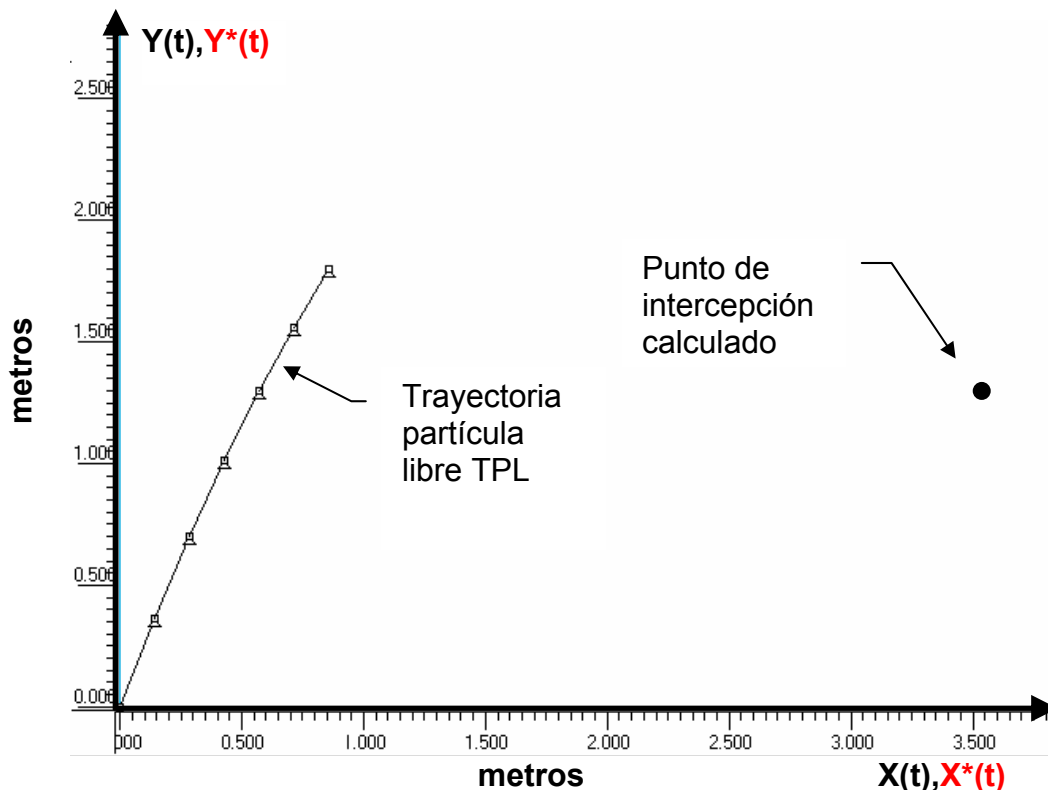
En la figura 3.1b, se muestra la segunda intercepción de la partícula libre 2, TPL2 en el instante de tiempo  $t_3 = 1.1$  s, se debe observar que la partícula libre 1 TPL1 ya fue interceptada, y que la continuación de su trayectoria es solo virtual, y se muestra con fines ilustrativos. El simulador también permite cambiar en forma rápida los parámetros, como son la posición de la partícula seguidora TPS antes de iniciar la intercepción, los ángulos de las partículas libres TPL1 y TPL2, así como también sus velocidades iniciales. También los tiempos de intercepción se pueden controlar y se van monitoreando y guardando los datos para luego realizar las gráficas correspondientes de posición y velocidad de las partículas.

### III.3 SIMULACIÓN COMPUTACIONAL DEL CASO I.

Los datos para realizar el evento físico de la figura 3.2a, 3.2b y 3.3 fueron tomados de [3.1], donde la velocidad máxima de lanzamiento de un objeto por un ser humano es de 10 m/s aprox. Y la mínima oscila entre 3.6 y 4 m/s. Por lo tanto, se puede tomar cualquier valor entre este rango incluyendo el ángulo de disparo que va desde los  $30^\circ$  y los  $87^\circ$ . Los datos son los siguientes;

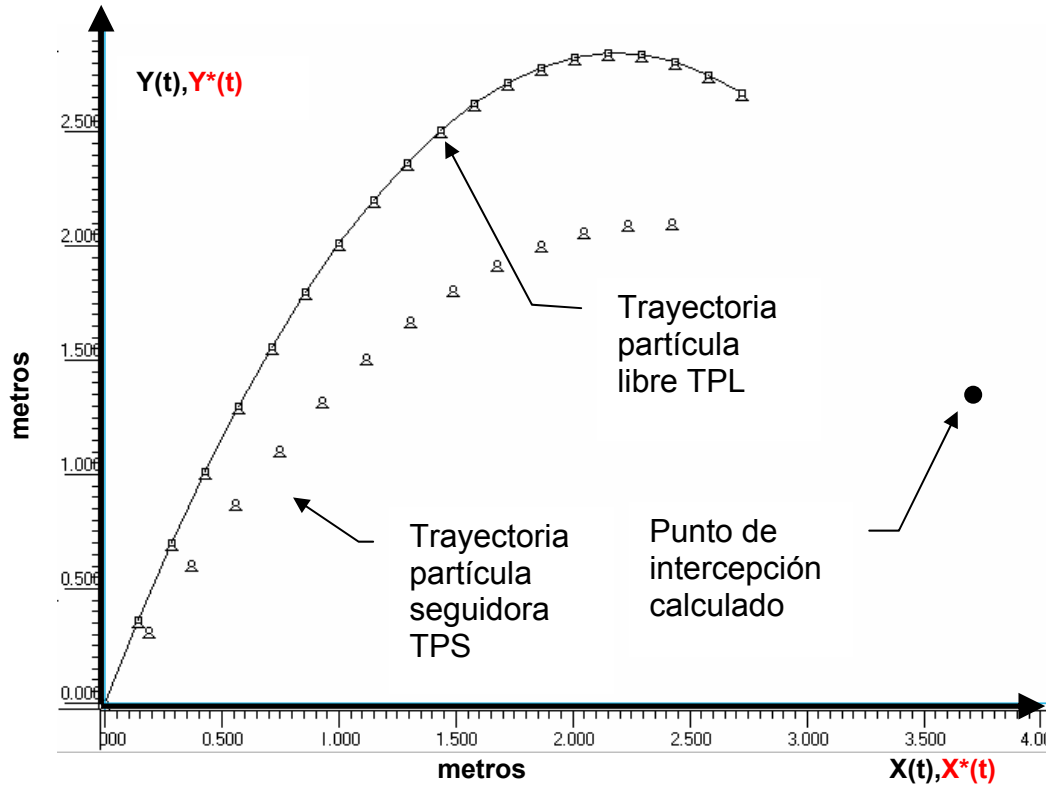
Velocidad inicial:	$ V_0  =$	8 m/s.
Ángulo de disparo:	$\theta =$	69° grados.
Tiempo de retardo:	$t_1 =$	0.3 seg.
Tiempo de intercepción:	$t_2 =$	1.3 seg.

A continuación se muestra una secuencia de dos imágenes del fenómeno de intercepción del caso I, con el fin de hacer ciertas observaciones:



**Figura 3.2a** Trayectorias de la partícula libre y la partícula seguidora, en el instante de tiempo  $t = 0.30$  s.

En la figura 3.2a se ilustra el movimiento de la partícula libre TPL en el tiempo absoluto  $t = 0.3$  s. Es importante mencionar que la partícula seguidora, en el tiempo  $t = 0.3$  s, que es el tiempo de retardo  $t_1 = 0.3$  s, empezará a moverse. En ese mismo instante la partícula libre TPL se encuentra a una distancia  $|d| = 1.994$  m en la posición  $(0.860, 1.799)$  del origen del sistema de coordenadas y con una velocidad de  $|V| = 5.358$  m/s en la posición  $(2.867, 4.527)$ , cuando la partícula seguidora inicia su movimiento en la posición de origen con una velocidad de 7.581 m/s.

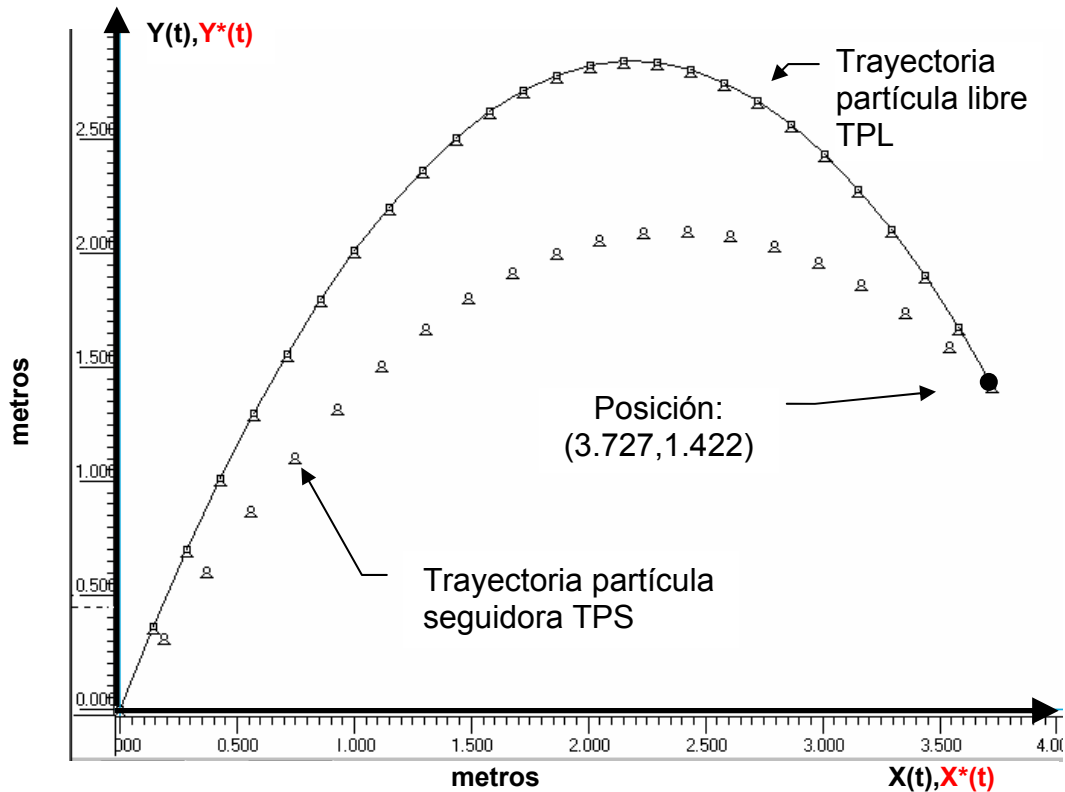


**Figura 3.2b** Trayectorias de la partícula libre y la partícula seguidora, en el instante  $t = 0.95$  s.

Ahora se analizará el evento en el tiempo absoluto  $t = 0.95$  s, con fines ilustrativos. Por lo tanto, de la figura 3.2b se obtuvieron los siguientes resultados mostrados en la tabla 3.1a, que corresponden al momento mostrado en la figura 3.2b.

<b>Tabla 3.1a</b>		
Resultados de la cinemática de la intersección de una partícula libre por una partícula seguidora en $t = 0.95$ seg.		
	Partícula libre TPL	Partícula seguidora TPS
Posición: m	$X(t) = 2.724$	$X^*(t) = 2.423$
	$Y(t) = 2.670$	$Y^*(t) = 2.103$
	$ d  = 3.814$	$ d  = 3.208$
Velocidad: m/s	$V_x(t) = 2.867$	$V_x^*(t) = 3.727$
	$V_y(t) = -1.848$	$V_y^*(t) = -0.131$
	$ V  = 3.411$	$ V ^* = 3.729$
Aceleración: $m/s^2$	$A_x = 0.0$	$A_x^* = 0.0$
	$A_y = -9.81$	$A_y^* = -10.359$

En la figura 3.3 se muestra finalmente en el tiempo de intersección  $t = 1.3$  s, y las trayectorias de las dos partículas que describieron en el evento.

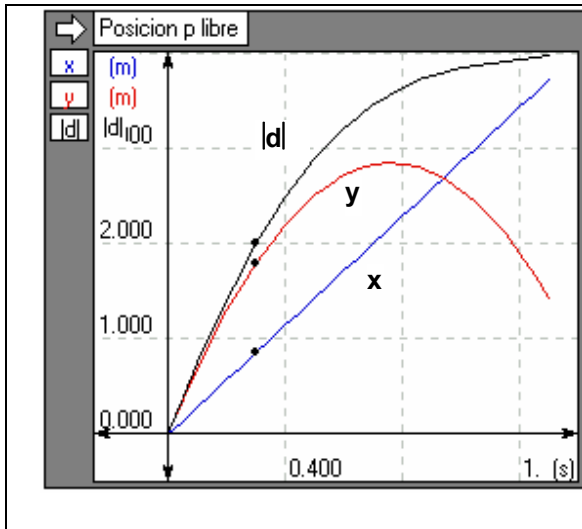


**Figura 3.3** Trayectorias de la partícula libre y la partícula seguidora, en el instante  $t_2 = 1.30$  seg. La posición en la intercepción de la partícula libre es:  $X(t_2) = 3.727$  y  $Y(t_2) = 1.422$  m.

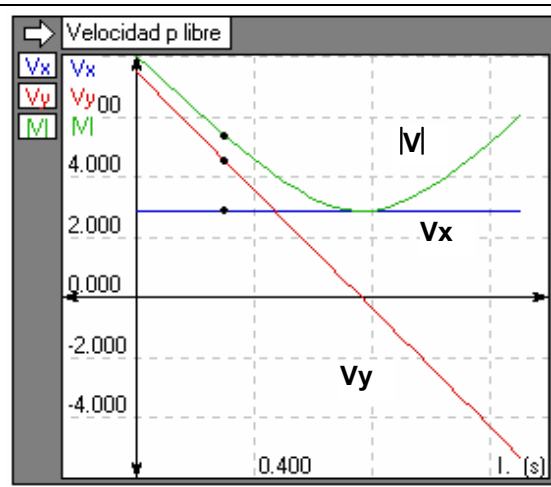
En la figura 3.3, se puede observar que la velocidad de la partícula libre en el instante de intercepción  $t_2=1.3$  s, es de  $|V| = 6.009$  m/s y la velocidad de la partícula seguidora de  $|V^*| = 5.292$  m/s lo que significa que existe un pequeño impacto debido a la diferencia de velocidades. La magnitud de la velocidad relativa es de  $|V| - |V^*| = 0.717$ . Es decir, existe impacto, la trayectoria seguidora golpea a la partícula libre al interceptarla. En la tabla 3.1b se registran los resultados numéricos de la simulación de la figura 3.3.

<b>Tabla 3.1b</b>		
Resultados de la cinemática de la intercepción de una partícula libre por una partícula seguidora a 1.3 s.		
	Partícula libre TPL	Partícula seguidora TPS
Posición: m	$X(t) = 3.727$	$X^*(t) = 3.727$
	$Y(t) = 1.422$	$Y^*(t) = 1.422$
	$ d  = 3.989$	$ d  = 3.989$
Velocidad: m/s	$V_x(t) = 2.867$	$V_x^*(t) = 3.727$
	$V_y(t) = -5.280$	$V_y^*(t) = -3.757$
	$ V  = 6.009$	$ V^*  = 5.292$
Aceleración: $m/s^2$	$A_x = 0.0$	$A_x^* = 0.0$
	$A_y = -9.81$	$A_y^* = -10.359$

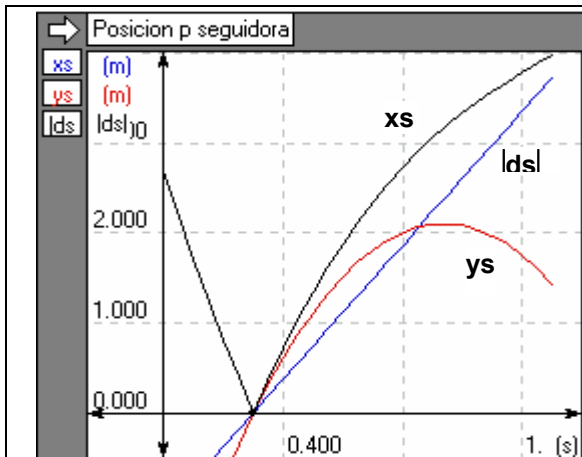
A continuación se presentan las gráficas del evento antes ilustrado ver la figura 3.2a.



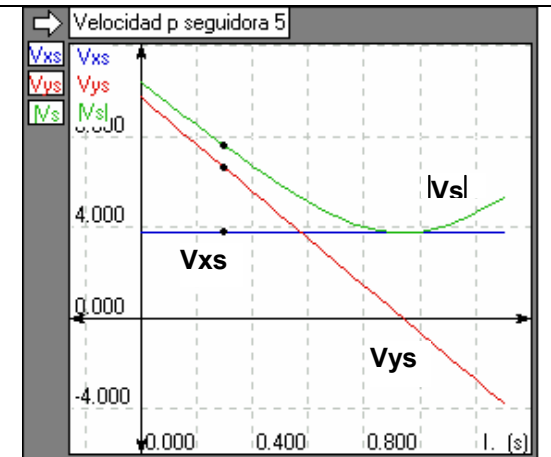
**Figura 3.4a** Gráfica posición-tiempo, en el instante  $t = 0.3$  s, que corresponde al retardo  $t_1$ . Partícula Libre.



**Figura 3.4b** Gráfica velocidad-tiempo, en el instante  $t = 0.3$  s, que corresponde al retardo  $t_1$ . Partícula Libre.



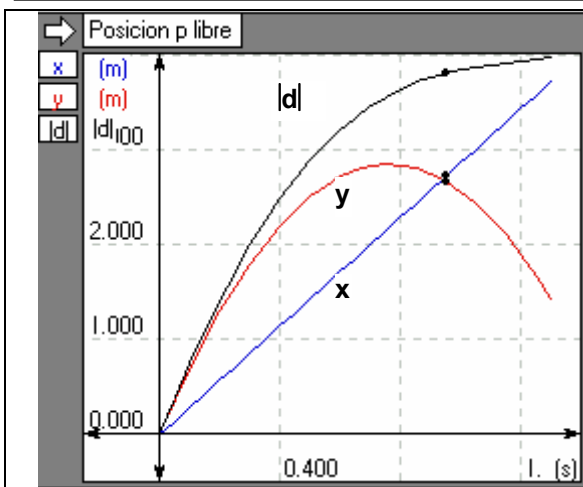
**Figura 3.4c** Gráfica posición-tiempo, en el instante  $t = 0.3$  s, que corresponde al retardo  $t_1$ . Partícula Seguidora.



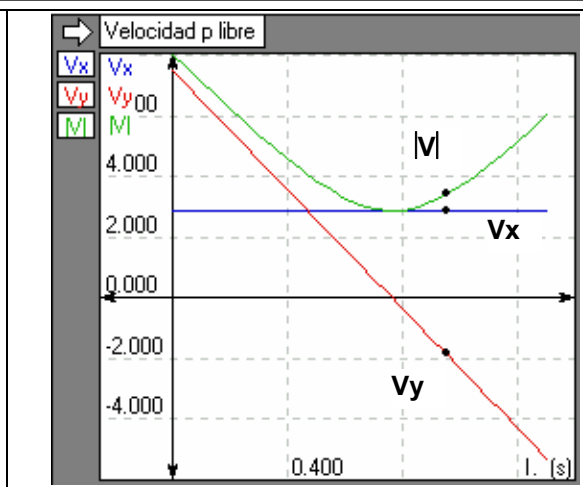
**Figura 3.4d** Gráfica velocidad-tiempo, en el instante  $t = 0.3$  s, que corresponde al retardo  $t_1$ . Partícula Seguidora.

Las aceleraciones para la partícula libre son:  $A_x = 0.00 \text{ m/s}^2$ ,  $A_y = -9.807 \text{ m/s}^2$  y la magnitud  $|A| = 9.807 \text{ m/s}^2$ . y para la partícula seguidora  $A_{xs} = 0.00 \text{ m/s}^2$ ,  $A_{ys} = -10.359 \text{ m/s}^2$  y la magnitud  $|A_s| = 10.359 \text{ m/s}^2$ .

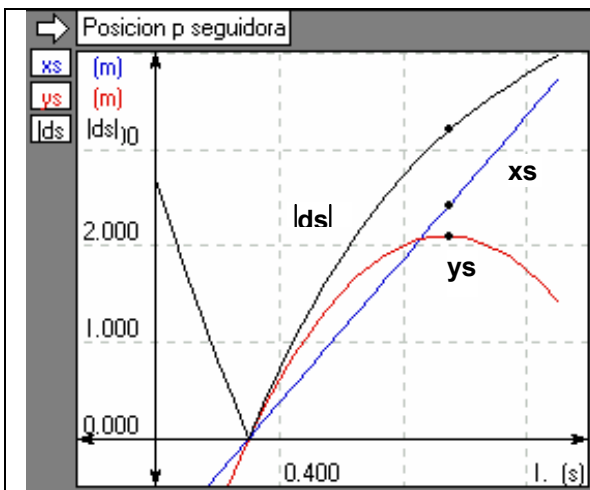
Ahora se presentan las gráficas del evento antes ilustrado ver la figura 3.2b. en donde se analiza el evento en el tiempo  $t=0.95$  s. El objeto de presentar estos tiempos es debido a que corresponden al tiempo de retardo y al tiempo de intercepción y a un instante intermedio para ilustrar el problema.



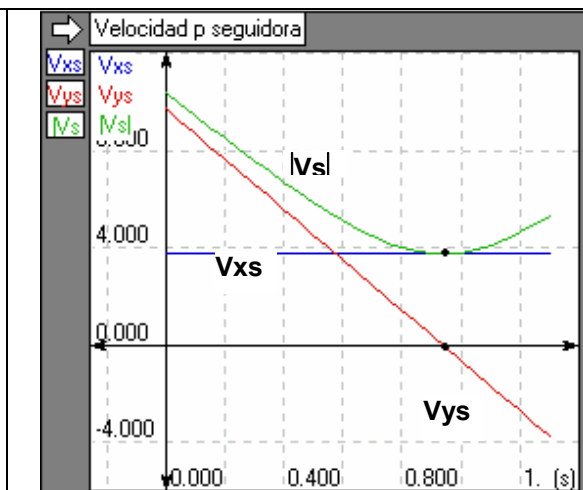
**Figura 3.4e** Gráfica posición-tiempo, en el instante  $t = 0.95$  s. Partícula Libre.



**Figura 3.4f** Gráfica velocidad-tiempo, en el instante  $t = 0.95$  s. Partícula Libre.

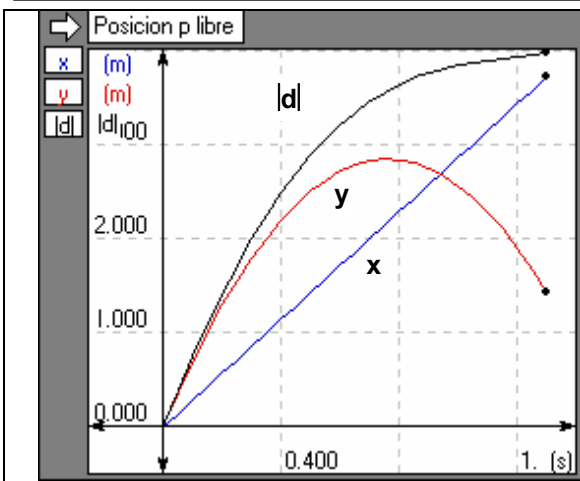


**Figura 3.4g** Gráfica posición-tiempo, en el instante  $t = 0.95$  s. Partícula Seguidora.

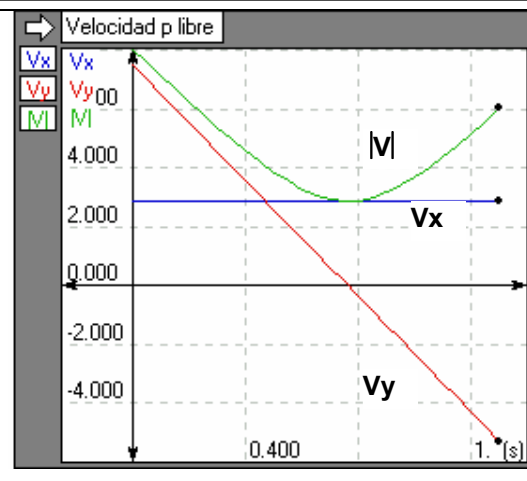


**Figura 3.4h** Gráfica velocidad-tiempo, en el instante  $t = 0.95$  s. Partícula Seguidora.

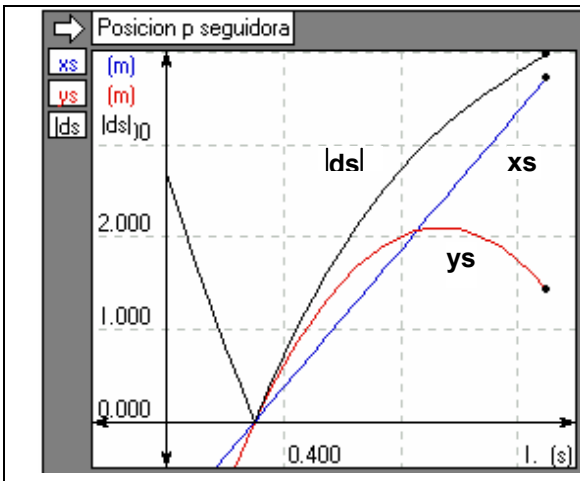
Finalmente se presentan las gráficas del evento antes ilustrado ver la figura 3.3. En donde se analiza el evento en el tiempo  $t=1.3$  s, que corresponde a la intercepción.



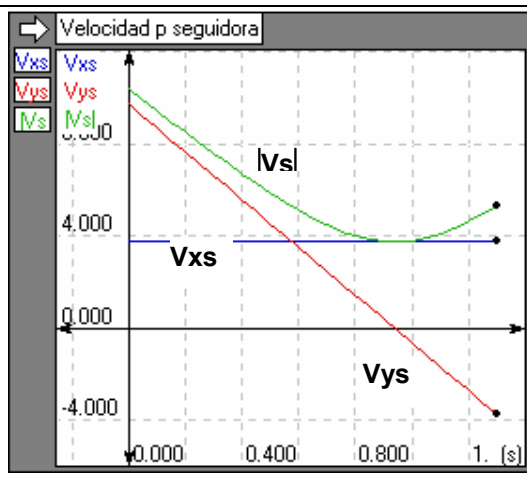
**Figura 3.4i** Gráfica posición-tiempo, en el instante  $t = 1.30$  s, que corresponde al tiempo de intercepción  $t_2$ . Partícula Libre.



**Figura 3.4j** Gráfica velocidad-tiempo, en el instante  $t = 1.30$  s, que corresponde al tiempo de intercepción  $t_2$ . Partícula Libre.



**Figura 3.4k** Gráfica posición-tiempo, en el instante  $t = 1.30$  s, que corresponde al tiempo de intercepción  $t_2$ . Partícula Seguidora.



**Figura 3.4l** Gráfica velocidad-tiempo, en el instante  $t = 1.30$  s, que corresponde al tiempo de intercepción  $t_2$ . Partícula Seguidora.

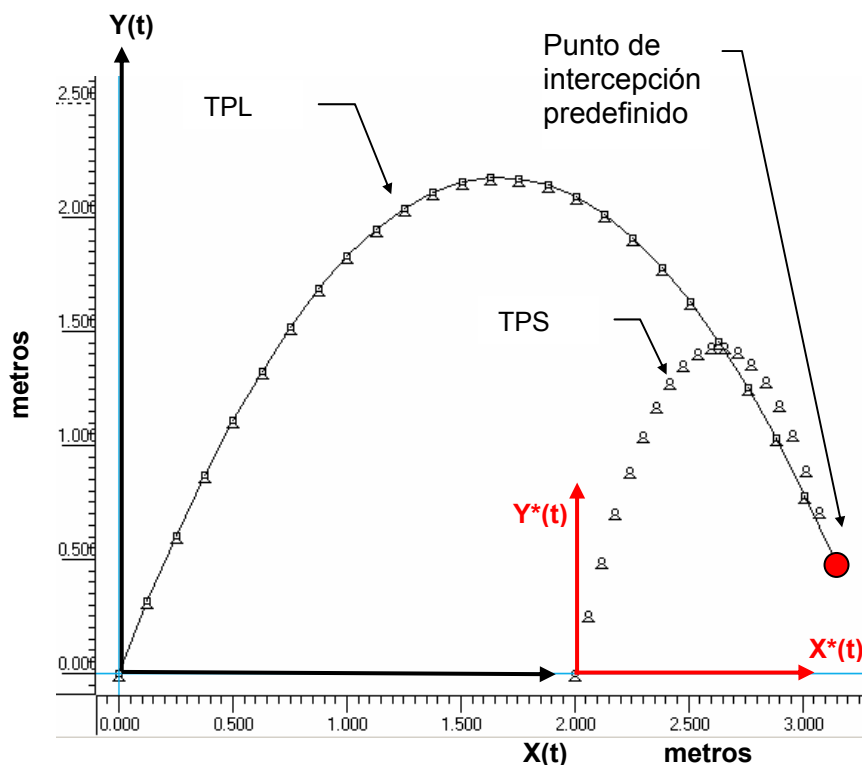
En las gráficas de las figuras 3.4 a...l, se observa que existe una distancia y una velocidad dadas de la partícula libre, cuando la posición de la partícula seguidora es cero, aunque su velocidad inicial es diferente de cero.

### III.4 SIMULACIÓN DEL CAMBIO DE BASE DEL CASO I.

En esta simulación se sustituirá sólo el componente  $X^*(t)$ , es decir, en la función de intercepción  $X^*(t)$  se incluye el término “a” que permite desplazar el origen de coordenadas de este eje de donde se iniciará el movimiento de la partícula seguidora. El componente  $Y^*(t)$  queda sin modificación. En otras palabras, el punto de inicio del movimiento de la partícula de intercepción se desplaza a la derecha dos metros con respecto al sistema de coordenadas, mostrado en la figura 3.5.

Los datos para realizar el evento de la figura 3.5 fueron los siguientes: Cabe mencionar, que estos datos son diferentes con respecto a los datos presentados en la página 79. lo anterior es con el fin de ir ilustrando diferentes eventos para validar la metodología propuesta.

Velocidad inicial:	$ V_0  =$	7 m/s.
Ángulo de disparo:	$\theta =$	69° grados.
Tiempo de retardo:	$t_1 =$	0.3 seg.
Tiempo de intercepción:	$t_2 =$	1.25 seg.
Cambio de base eje $x^*(t)$	$a =$	2.0 m

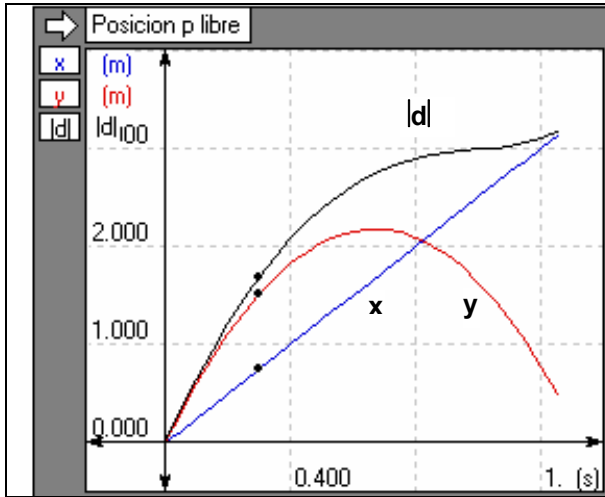


**Figura 3.5.** Trayectorias de la partícula libre y la partícula seguidora en el tiempo  $t_2 = 1.25$  s. La posición en el tiempo de intercepción es  $X(t_2) = 3.136$  y  $Y(t_2) = 0.507$  m. y  $a = 2$  m de distancia desde el origen

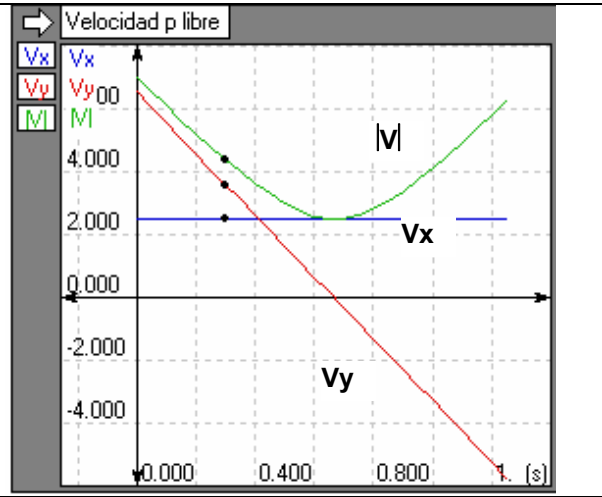
Nuevamente, en la figura 3.5 se puede observar que la magnitud de la velocidad de la partícula libre en el instante de intercepción  $t_2 = 1.25$  s es de  $|V| = 6.249$  m/s, y la velocidad de la partícula seguidora es de  $|V_s| = 4.568$  m/s, lo que significa que existe un



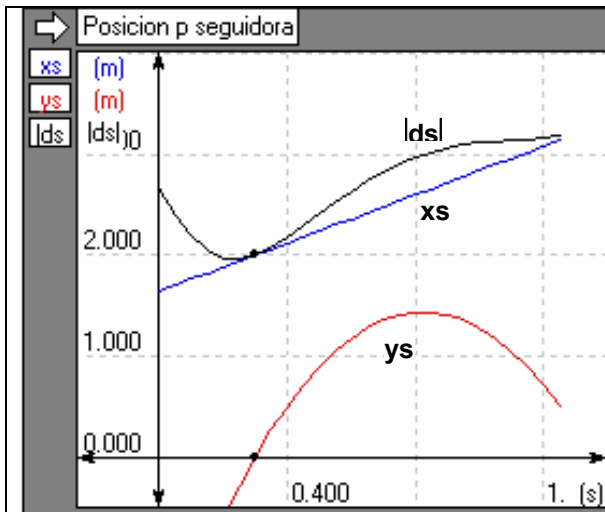
pequeño impacto debido a la diferencia de velocidades. La magnitud de la velocidad relativa de las dos partículas es de  $|V| - |V_s| = 1.681 \text{ m/s}$ . A continuación se presentan las siguientes gráficas generadas del evento antes ilustrado:



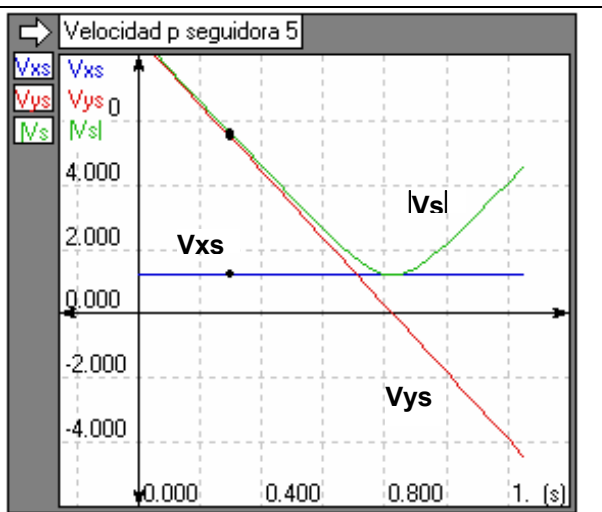
**Figura 3.6a** Gráfica posición-tiempo en el instante  $t= 0.3 \text{ s}$ , que corresponde al retardo  $t_1$ . Partícula libre.



**Figura 3.6b** Gráfica velocidad-tiempo en el instante  $t= 0.3 \text{ s}$ , que corresponde al retardo  $t_1$ . Partícula libre.



**Figura 3.6c** Gráfica posición-tiempo en el instante  $t= 0.3 \text{ s}$ , que corresponde al retardo  $t_1$ . Partícula seguidora.



**Figura 3.6d** Gráfica velocidad-tiempo en el instante  $t= 0.3 \text{ s}$ , que corresponde al retardo  $t_1$ . Partícula seguidora.

Las aceleraciones de las partículas son:  $A_x = 0.00$ ,  $A_y = -9.807 \text{ m/s}^2$ ,  $|A| = 9.807 \text{ m/s}^2$ ,  $A_{x_s} = 0.00$ ,  $A_{y_s} = -10.406 \text{ m/s}^2$ ,  $|A_s| = 10.406 \text{ m/s}^2$ .

En las figuras 3.6 a-d, se puede observar que existe una distancia y una velocidad dadas de la partícula libre cuando la posición de la partícula seguidora es cero, aunque la velocidad inicial de esta última es diferente de cero.

## III.5 SIMULACIÓN DE LA INTERCEPCIÓN MÚLTIPLE DE 2 PARTÍCULAS LIBRES.

En esta simulación se sustituye nuevamente sólo el componente  $X^*(t)$ , el componente  $Y^*(t)$  queda sin modificación, debido a que se supone que la partícula seguidora TPS saldrá de la línea horizontal desde la cual las partículas libres TPL1 y TPL2 inician su movimiento. Los datos para realizar el evento de la figura 3.7 fueron los siguientes:

Partícula # 1 TPL1		
Velocidad inicial:	$ V_0  =$	6.5 m/s.
Ángulo de disparo:	$\theta_0 =$	80° grados.
Tiempo de retardo:	$t_1 =$	0.3 seg.
Tiempo de intercepción:	$t_2 =$	0.6 seg.

Partícula # 2 TPL2		
Velocidad inicial:	$ V_1  =$	7 m/s.
Ángulo de disparo:	$\theta_1 =$	102° grados.
Tiempo de retardo:	$t_1 =$	0.3 seg.
Tiempo de intercepción:	$t_3 =$	1.1 seg.

Los tiempos de intercepción se fijan arbitrariamente dependiendo de las condiciones del evento de intercepción.

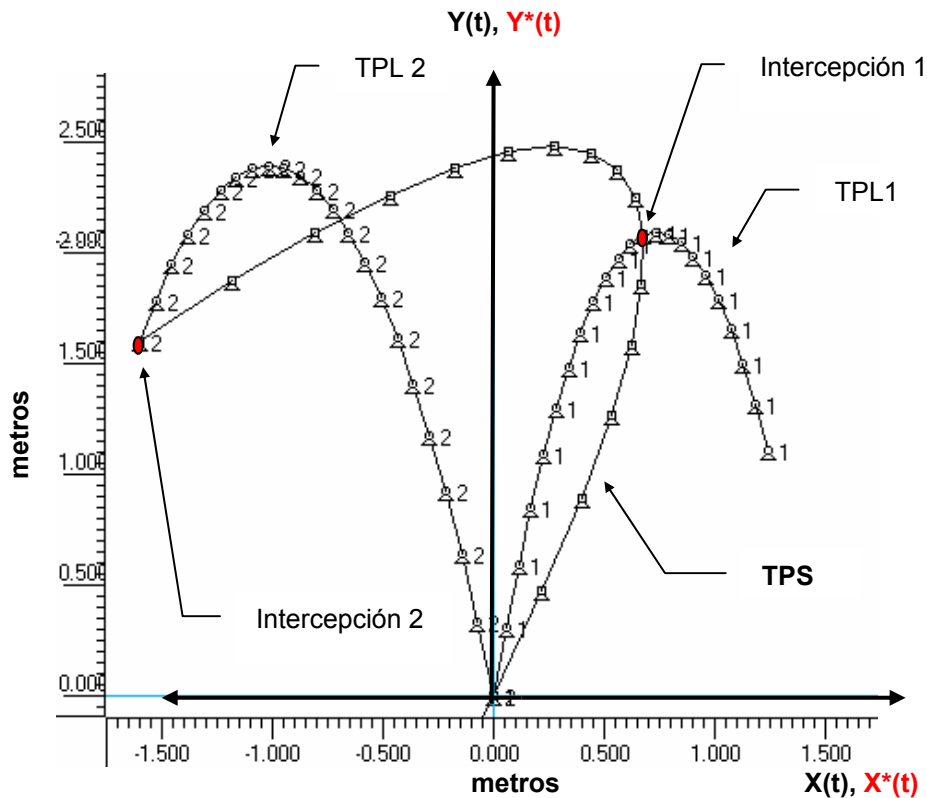
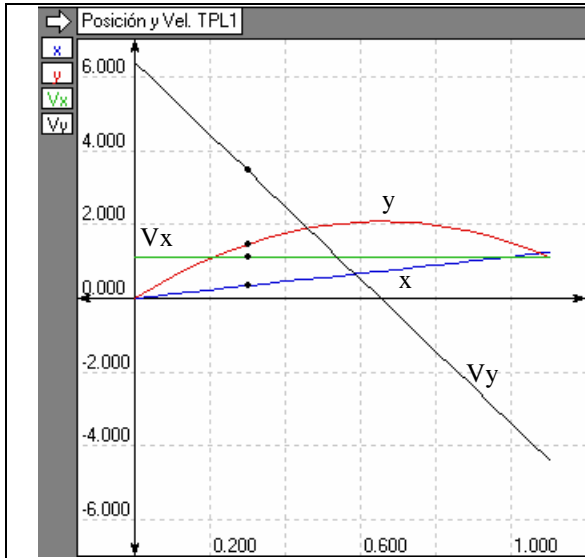
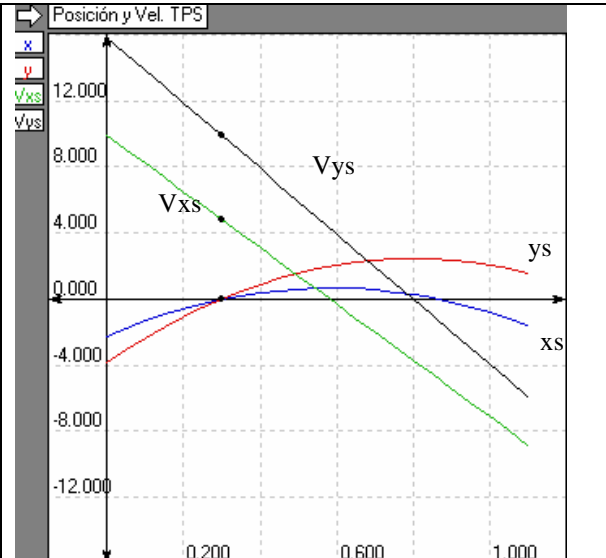


Figura 3.7. La trayectoria TPL1 es interceptada primero por la trayectoria TPS, después intercepta la trayectoria TPL2.

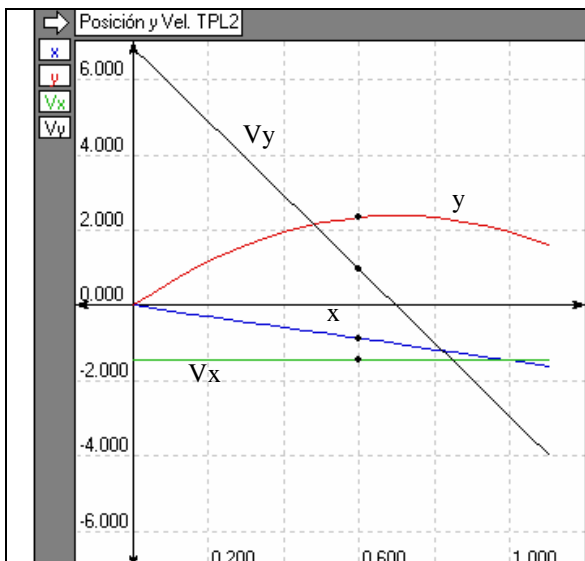
Las observaciones para la partícula libre TPL1 son: En el instante  $t=0.6$  s, la partícula seguidora TPS la intercepta en la posición  $(0.677, 2.075)$ , con una velocidad de  $|V| = 1.242$  m/s. La partícula TPS en ese mismo instante tiene una velocidad de  $|Vs| = 3.977$ . Por lo tanto, la partícula TPS golpea a la partícula TPL1 con una velocidad de 2.735 m/s. Este detalle del impacto se elimina en la modificación de la metodología para evitar el impacto más adelante.



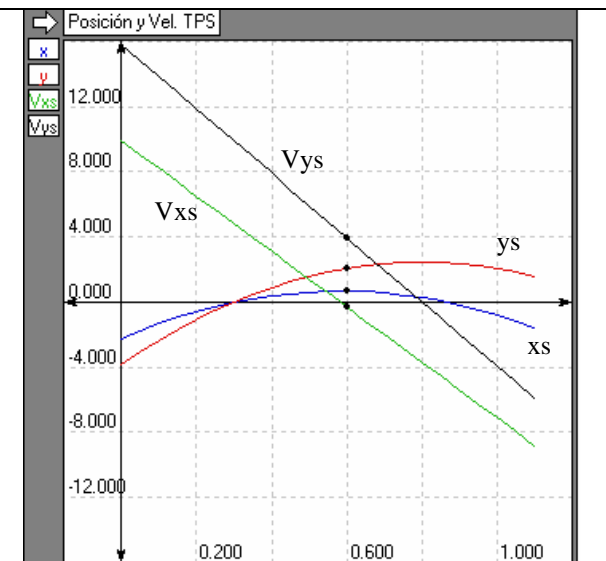
**Figura 3.8a** Gráfica posición-tiempo y velocidad-tiempo en el instante  $t= 0.3$  s. que corresponde al retardo  $t_1$ . Partícula libre.



**Figura 3.8b** Gráfica posición-tiempo y velocidad-tiempo en el instante  $t= 0.3$  s. de retardo, que corresponde al retardo  $t_1$ . Partícula seguidora.



**Figura 3.8c** Gráfica posición-tiempo y velocidad-tiempo en el instante  $t= 0.6$  s, que corresponde al tiempo de intercepción 1, tiempo  $t_2$ . Partícula libre.



**Figura 3.8d** Gráfica posición-tiempo y velocidad-tiempo en el tiempo  $t= 0.6$  s, que corresponde al tiempo de intercepción 1, tiempo  $t_2$ . Partícula seguidora.

Las observaciones para la partícula libre TPL2 son: en el instante  $t=1.1$  s, la partícula seguidora TPS intercepta a TPL2 que se encuentra ya en la posición de  $(-1.601, 1.599)$  y con una velocidad de  $|V| = 4.201$  m/s. La partícula TPS en ese mismo instante tiene una velocidad de  $|V_s| = 10.593$ . Por lo tanto, la partícula TPS golpea a la partícula TPL1 con una velocidad de  $6.392$  m/s. Nuevamente existe impacto, esta colisión en otras aplicaciones puede ser desfavorable, por lo tanto más adelante se modifica la metodología para eliminar este problema.

Las aceleraciones son:  $A_x = 0.00$ ,  $A_y = -9.807$  m/s<sup>2</sup>,  $|A| = 9.807$  m/s<sup>2</sup>,  $A_{x_s} = -17.0343$  m/s<sup>2</sup>,  $A_{y_s} = -19.6807$  m/s<sup>2</sup>,  $|A_s| = 26.0288$  m/s<sup>2</sup>.

En las gráficas anteriores, se puede observar que existe una distancia y una velocidad dadas de la partícula libre cuando la posición de la partícula seguidora es cero, aunque la velocidad inicial de ésta es diferente de cero.

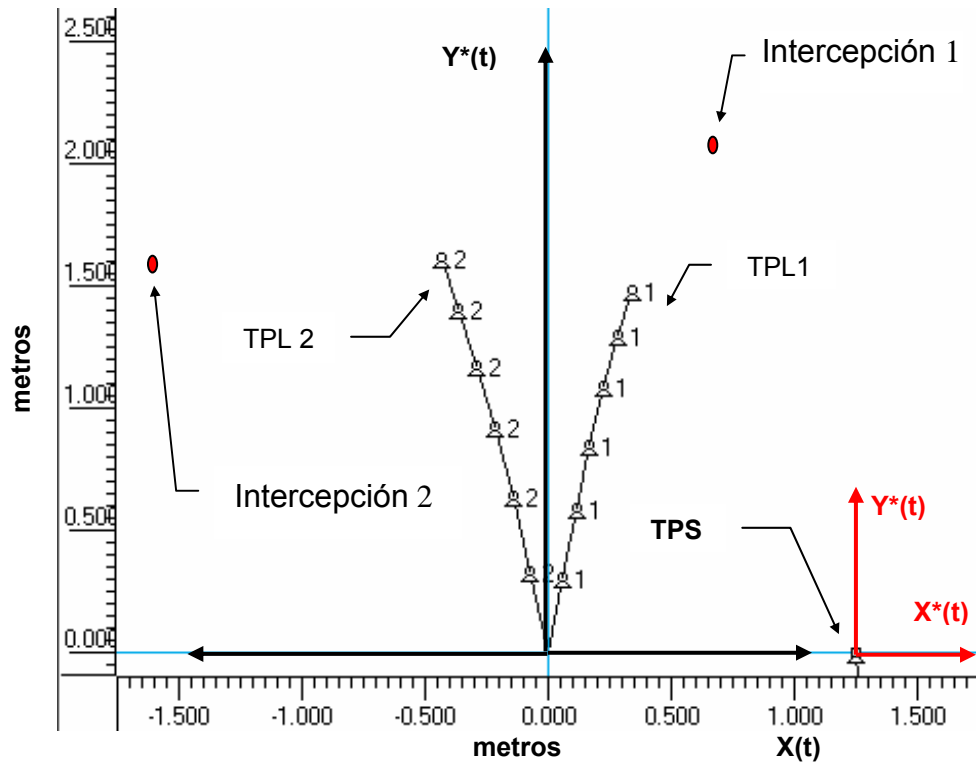
### III.6 SIMULACIÓN INTERCEPCIÓN MULTIPLE DE 2 PARTÍCULAS CAMBIO DE BASE.

En esta simulación se considera sólo la componente de la posición, velocidad y aceleración de  $X^*(t)$ . Es decir, el componente  $Y^*(t)$  permanecerá inalterado. Los datos para realizar el evento de la figura 3.9, 3.10 y 3.11 fueron los siguientes:

Partícula # 1 TPL1		
Velocidad inicial:	$ V_0  =$	6.5 m/s.
Ángulo de disparo:	$\theta_0 =$	80° rados.
Tiempo de retardo:	$t_1 =$	0.3 seg.
Tiempo de intercepción:	$t_2 =$	0.6 seg.
Cambio de base eje $x^*(t)$	$a =$	1.25 m

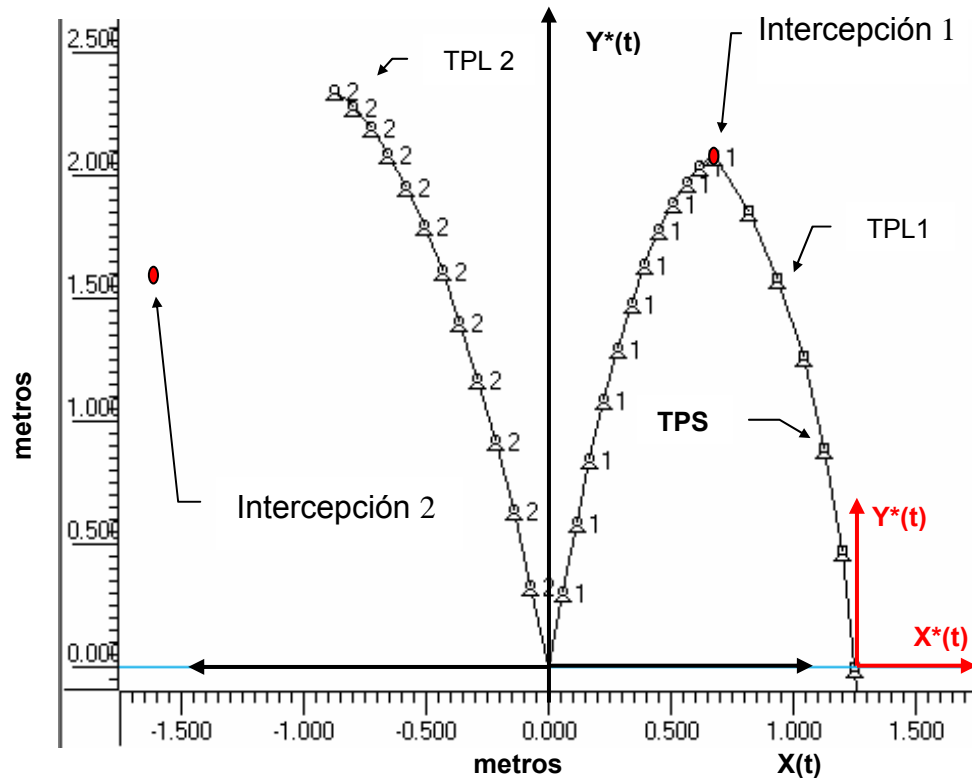
Partícula # 2 TPL2		
Velocidad inicial:	$ V_1  =$	7 m/s.
Ángulo de disparo:	$\theta_1 =$	102° grados.
Tiempo de retardo:	$t_1 =$	0.3 seg.
Tiempo de intercepción:	$t_3 =$	1.1 seg.
Cambio de base eje $x^*(t)$	$a =$	1.25 m

En la figura 3.9, se muestra el instante de tiempo  $t = 0.3$  s, la posición de la partícula TPL1 es  $(0.339, 1.479)$  y su velocidad es de  $|V| = 3.639$  m/s. La partícula TPL2 se encuentra en la posición  $(-0.437, 1.613)$  y su velocidad es de  $|V| = 4.167$  m/s. La partícula TPS está en  $(1.25, 0)$  e inicia su movimiento con una velocidad de  $|V_s| = 9.913$  m/s.



**Figura 3.9** Trayectorias TPL1 y TPL2 en el tiempo  $t = 0.3$  s, tiempo de retardo  $t_1$ .

En la figura 3.10, en el instante  $t = 0.6$  s, la posición de la partícula TPL1 y TPS es  $(0.677, 2.075)$  y su velocidad es de  $|V| = 1.242$  m/s y de  $|V_s| = 4.914$  m/s, respectivamente. La partícula TPL2 se encuentra en la posición  $(-0.873, 2.343)$  y su velocidad es de  $|V| = 1.745$  m/s.



**Figura 3.10** Trayectorias TPL1 y TPL2 en el tiempo  $t = 0.6$  s, tiempo de intercepción  $t_2$ .

En la figura 3.11, en el instante  $t = 1.1$  s, la posición de la partícula TPL1 sería virtualmente (1.242, 1.108) y su velocidad virtual sería de  $|V| = 4.529$  m/s. Por que ya fue interceptada en el tiempo  $t=0.6$  s, mientras tanto, la partícula TPL2 se encuentra en (-1.601, 1.599) y su velocidad es  $|V| = 4.201$  m/s. La partícula TPS en esta posición tiene una velocidad de  $|V_s| = 8.549$  m/s.

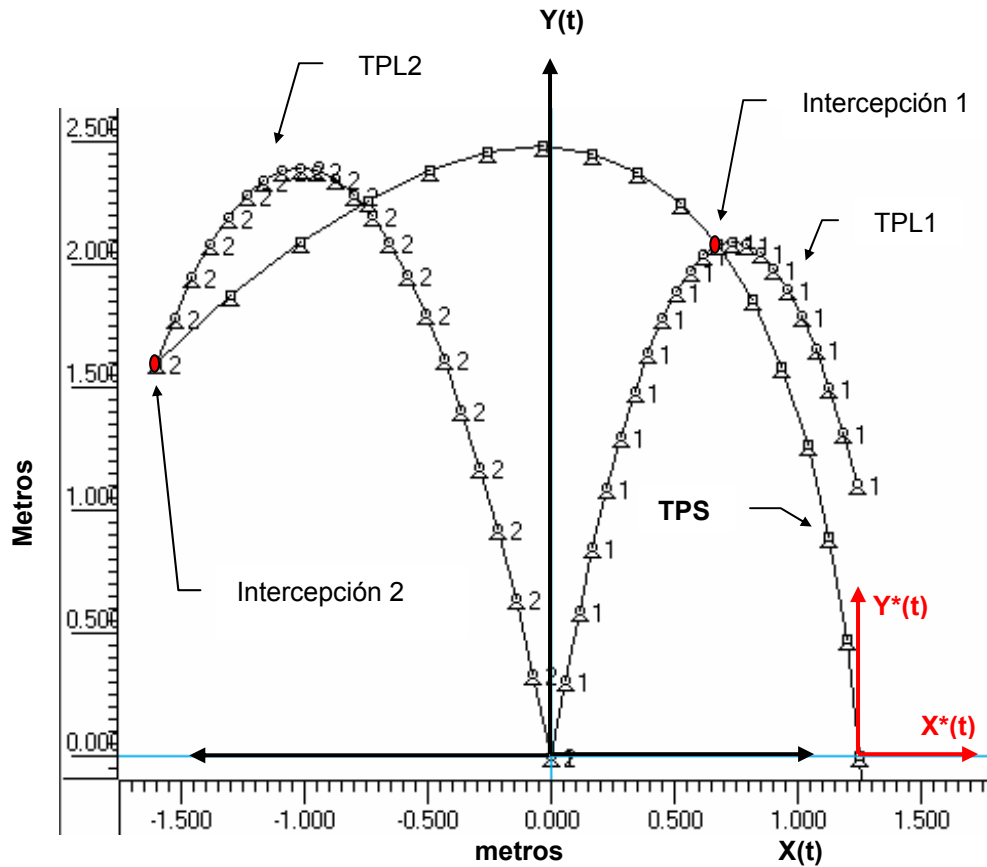
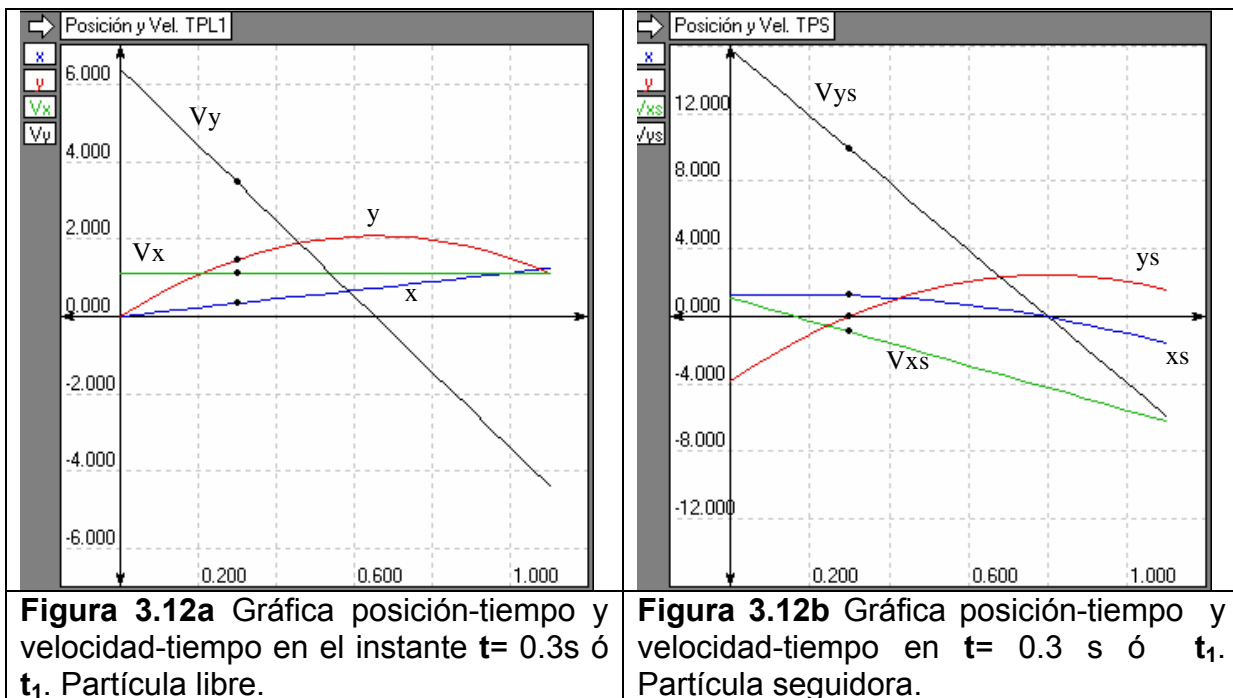
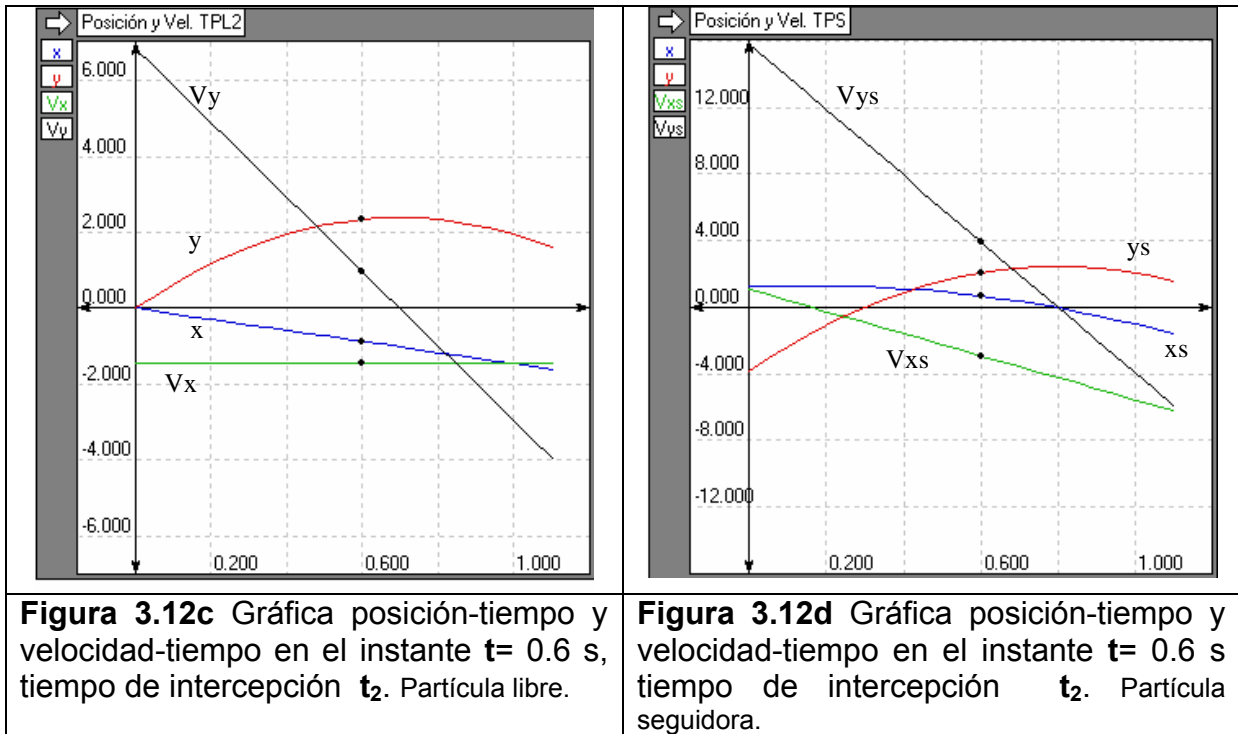


Figura 3.11 Trayectorias TPL1 y TPL2 en el tiempo  $t = 1.1$  s, tiempo de intercepción  $t_3$ .

A continuación se presentan las siguientes gráficas del evento antes ilustrado:



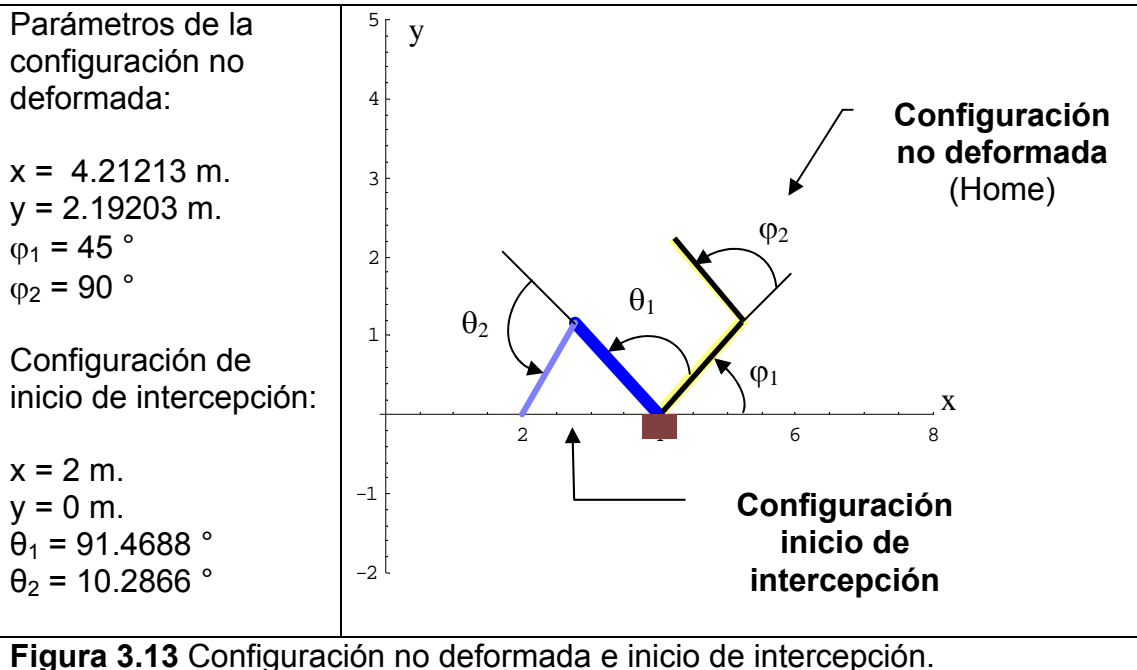


Las aceleraciones son:  $A_x = 0.00$ ,  $A_y = -9.807 \text{ m/s}^2$ ,  $|A| = 9.807 \text{ m/s}^2$ ,  $A_{x_s} = -6.61764 \text{ m/s}^2$ ,  $A_{y_s} = -19.6807 \text{ m/s}^2$ ,  $|A_s| = 26.0288 \text{ m/s}^2$ .

### III.7 CONFIGURACIÓN NO DEFORMADA Y CONFIGURACIÓN INICIO DE INTERCEPCIÓN DEL ROBOT INTERCEPTOR

La implementación de los algoritmos de la cinemática inversa de un robot manipulador de 2 gdl se muestra en la figura 3.13. Se utilizó el álgebra de Cuaterniones para obtener el modelo cinemático del robot interceptor. El listado del programa se encuentra en anexo A. Tiro\_parqcb.nb





En la figura 3.13 se muestra el robot interceptor de 2 gdl. Las longitudes de sus eslabones son:  $l_1 = 1.4$  m y  $l_2 = 1$  m, con juntas de revolución. La posición del efector final es (2,0) y la de la base del robot interceptor (4,0) referidos al sistema de coordenadas fijo. El origen del sistema de coordenadas se encuentra en (0,0), y es el punto donde se realiza el disparo de las partículas libres TPL1 y TPL2.

La configuración no deformada del robot interceptor es la posición de “Home”, desde esta posición se miden los desplazamientos angulares de los eslabones 1 y 2, como se muestran en la figura 3.13. Es muy importante mencionar, que el método de solución (Newton Raphson) que se aplica para obtener los desplazamientos angulares en la Cinemática Inversa, se obtienen desde la posición de configuración no deformada (Home) del robot interceptor. Por lo tanto, para obtener los desplazamientos relativos a la última posición alcanzada, es necesario realizar la diferencia de ángulos de ángulos.

En la configuración inicio de intercepción el robot está preparado para la tarea de intercepción. La posición puede ajustarse al área de trabajo conveniente.

### III.7.1 ANALISIS DE LOS CASOS I Y II PARA LA INTERCEPCIÓN DE UNA PARTÍCULA.

Se simulará la generación de las trayectorias seguidoras de los casos I y II, que se presentan con las ecuaciones 2.9, 2.10, 2.14 y 2.15. Lo anterior permitirá seleccionar el caso más adecuado. Para generar estas simulaciones se hizo uso del Software computacional de cálculo simbólico, Mathematica<sup>®</sup> 4.0.

$$\begin{aligned}
 X^*(t) &= 0, & 0 \leq t \leq t_1 \\
 X^*(t) &= \frac{1}{t_2 - t_1} \left( |V_0| \cos \theta t_2 (t - t_1) - a(t - t_2) \right), & t > t_1
 \end{aligned}
 \tag{3.1}$$

Caso I

$$\begin{aligned}
 Y^*(t) &= 0, & 0 \leq t \leq t_1 \\
 Y^*(t) &= \frac{1}{t_2 - t_1} \left[ |V_0| \sin \theta t_2 (t - t_1) - \frac{g t_2^2}{2(t_2^2 - t_1^2)} (t^2 - t_1^2) \right], & t \geq t_1
 \end{aligned}
 \tag{3.2}$$

Caso II

$$\begin{aligned}
 Y^*(t) &= 0, & 0 \leq t \leq t_1 \\
 Y^*(t) &= \frac{1}{t_2 - t_1} \left[ |V_0| \sin \theta t_2 (t - t_1) - \frac{g t_2^2}{2(t_2 - t_1)^2} (t - t_1)^2 \right], & t \geq t_1
 \end{aligned}
 \tag{3.3}$$

En esta simulación se sustituirá solo el componente  $X^*(t)$ , el componente  $Y^*(t)$  queda sin modificación. Los datos para realizar el evento de la figura 3.14 fueron los siguientes:

Partícula TPL		
Velocidad inicial:	$ V_0  =$	8 m/s.
Ángulo de disparo:	$\theta_0 =$	69° grados.
Tiempo de retardo:	$t_1 =$	0.3 seg.
Tiempo de intercepción:	$t_2 =$	1.2 seg.
Cambio de base eje $x^*(t)$	$a =$	2 m
Cambio de base eje $y^*(t)$	$b =$	0 m

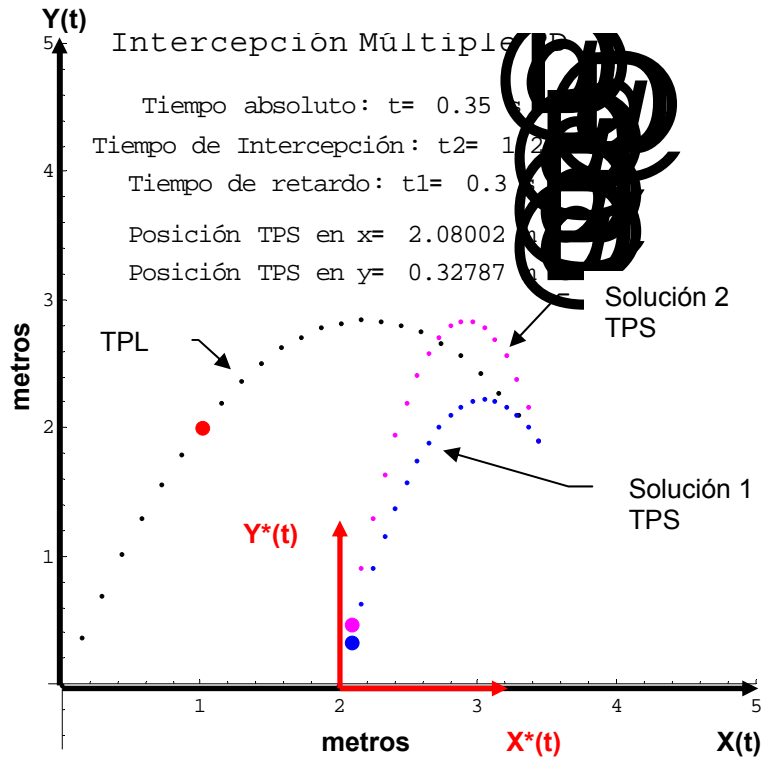


Figura 3.14 Soluciones del algoritmo de intercepción para una partícula libre.

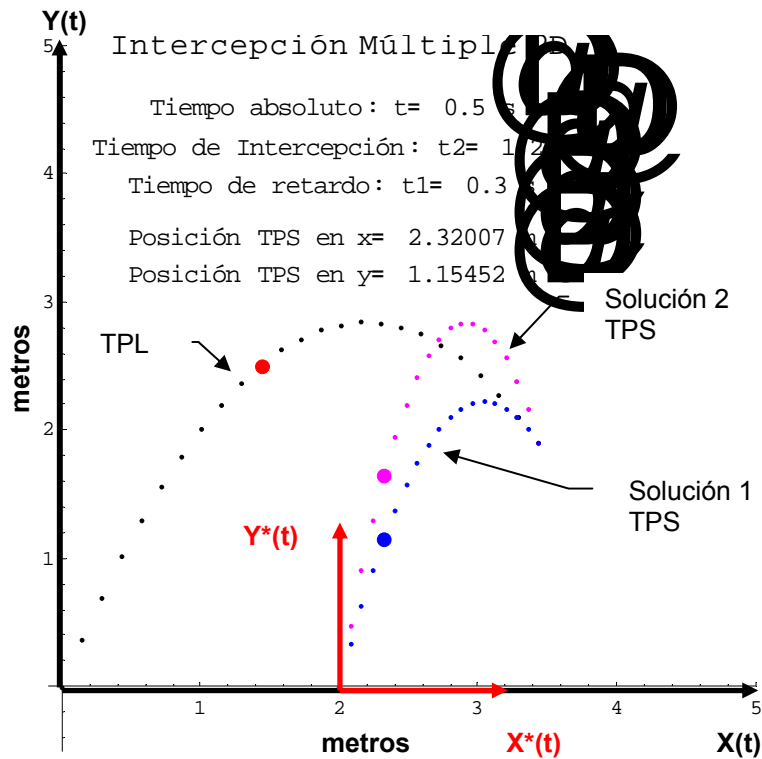
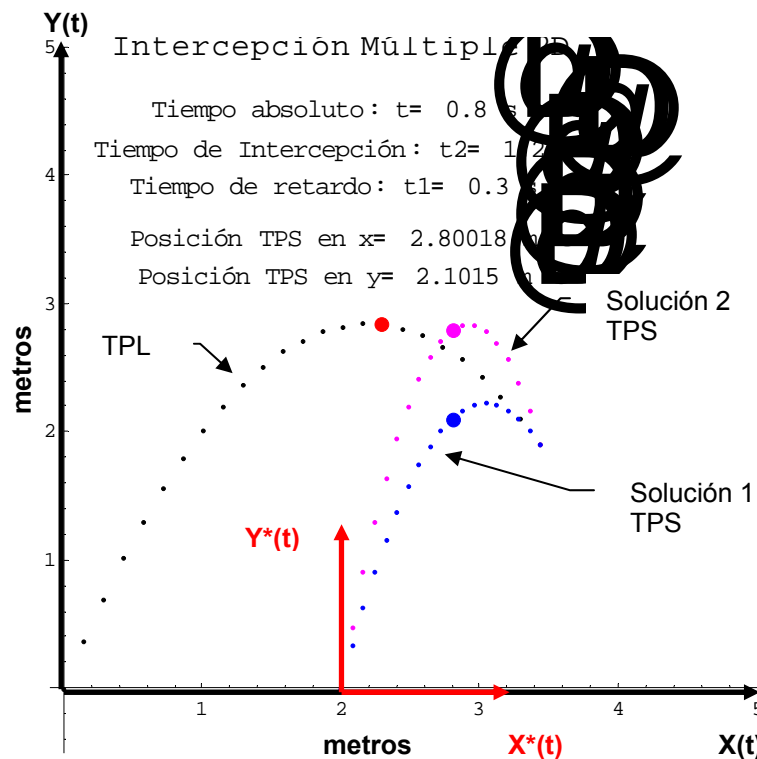


Figura 3.15 Soluciones del algoritmo de intercepción para una partícula libre.

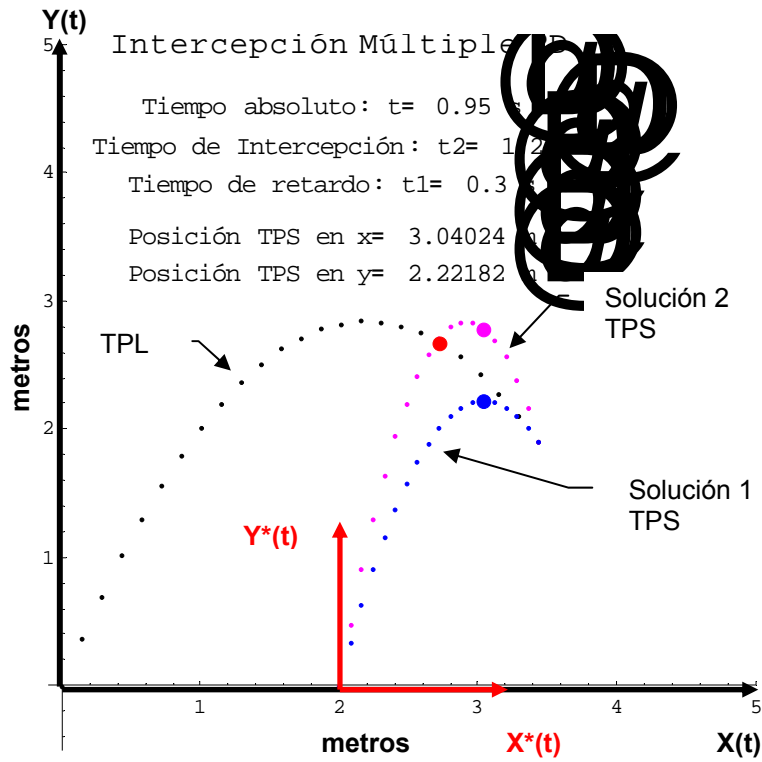
En la figura 3.14, se muestran las trayectorias seguidoras que generan las ecuaciones 3.1 y 3.2 en el instante de tiempo  $t = 0.35$  s, las soluciones se dan por el término cuadrático en la ecuación 1.1. Es decir, existen dos soluciones para interceptar la partícula libre TPL, La solución 1 TPS mostrada en la figura 3.14 presenta una trayectoria más corta comparada con la trayectoria de la solución 2 TPS en términos de desplazamiento lineal, medido desde el origen del sistema de coordenadas fijo. Este análisis permite determinar la trayectoria más corta que puede seguir el efector final del robot interceptor, pero también la orientación del efector final [3.2].

En la figura 3.15, se muestra el evento de intercepción por los casos I y II en el tiempo absoluto  $t = 0.5$  s, Es importante mencionar que la trayectoria descrita por la partícula libre TPL es conocida. También se puede observar que la velocidad de la Solución 2 TPS es mayor comparada con la velocidad de la Solución 1 TPS, esto es debido a que la segunda solución debe recorrer mayor distancia en el mismo tiempo.

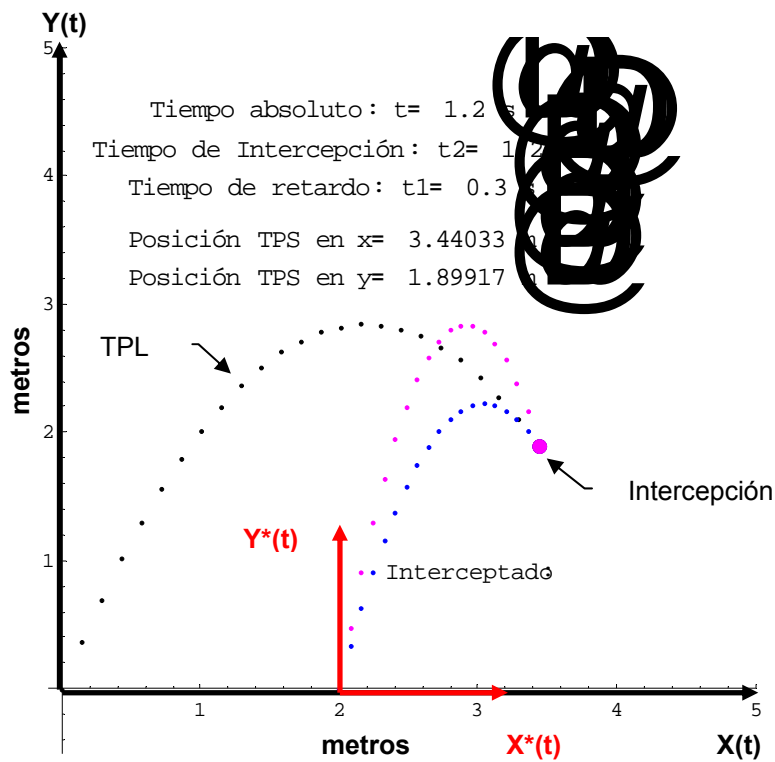


**Figura 3.16** Soluciones del algoritmo de intercepción para una partícula libre.

En la figura 3.16, se observa que la solución 2 TPS sigue a la partícula libre TPL tratando de alcanzarla en el tiempo de intercepción  $t_2$ . En cambio, la solución 1 TPS se adelanta a la trayectoria de la partícula libre para interceptarla en el mismo instante  $t_2$ . Es evidente que en ningún caso la velocidad relativa de la partícula libre TPL y la partícula seguidora TPS es cero. Lo anterior significa físicamente que existe impacto.



**Figura 3.17a** Soluciones del algoritmo de intercepción para una partícula libre.



**Figura 3.17b** Soluciones del algoritmo de intercepción para una partícula libre.

En las figuras 3.17a y 3.17b, se puede observar cuando las soluciones 1 y 2 interceptan la partícula libre TPL en el tiempo de intercepción  $t_2$ .

### III.8 INTERCEPCIÓN DE UNA PARTÍCULA LIBRE VÍA ROBOT MANIPULADOR DE 2 GDL.

Para realizar la implementación de los algoritmos de intercepción vía un robot manipulador de 2 gdl. Se adecuaron las trayectorias con los datos mostrados en la figura 3.16 con el objetivo de que se encuentren dentro del área de trabajo del robot. Es importante, mencionar que la posición óptima del robot para interceptar las partículas depende de las condiciones y restricciones físicas que están en el área de trabajo del robot y evitar los posibles obstáculos.

En la figura 3.18a se muestra la posición del robot interceptor en el instante  $t = 0.35$  s, el efector final seguirá la trayectoria seguidora TPS del caso I (solución 1 TPS). Debido a que la distancia es menor, logrando minimizar el movimiento del robot y aumentando el ahorro de energía para realizar la misma tarea de intercepción.

En las figuras 3.18b...d, solo muestran las posiciones en las cuales se realiza la simulación completa de intercepción.

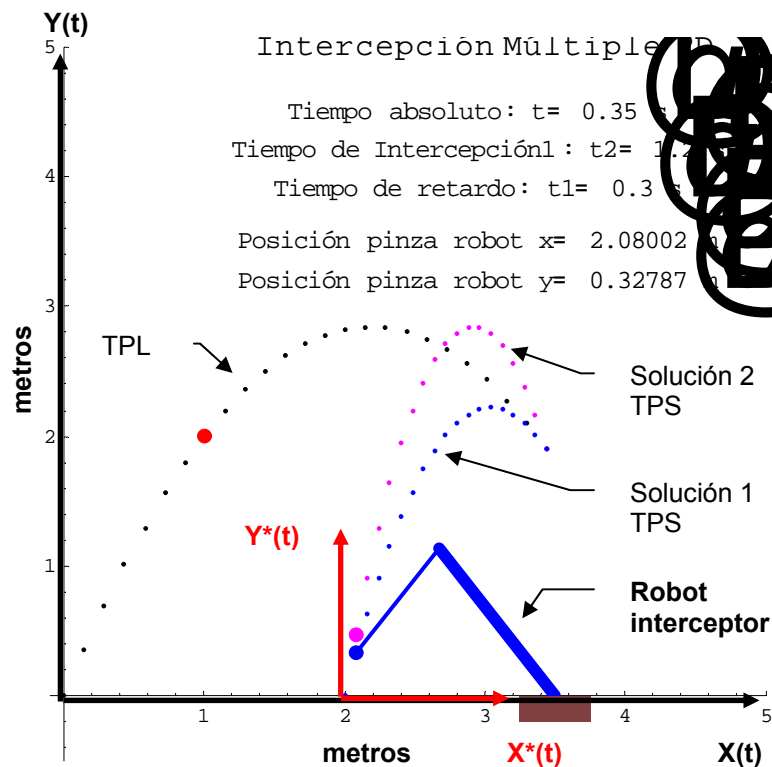


Figura 3.18a Intercepción de la partícula libre usando el caso I para una partícula libre.

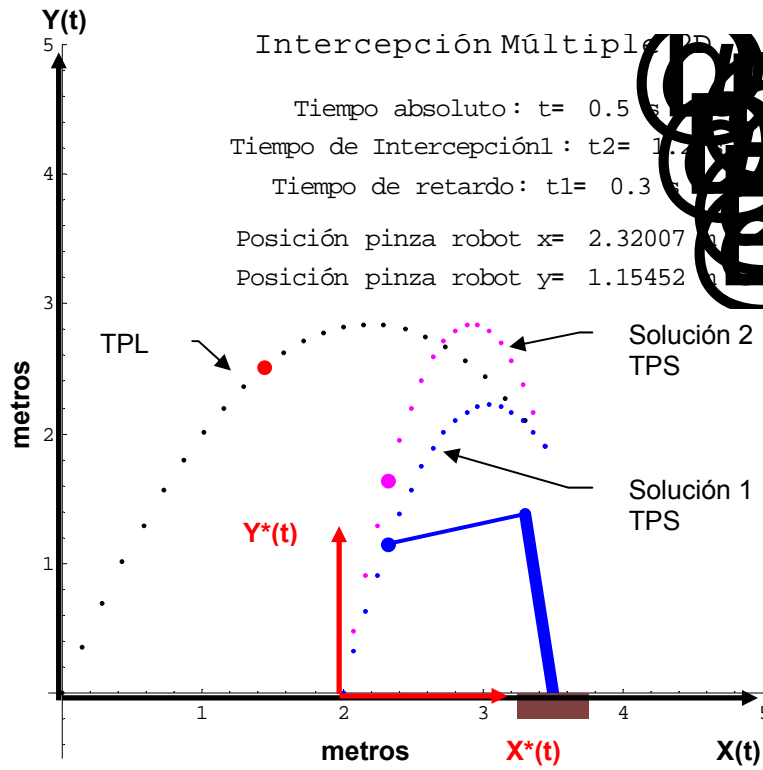


Figura 3.18b Intercepción de la partícula libre usando el caso I para una partícula libre.

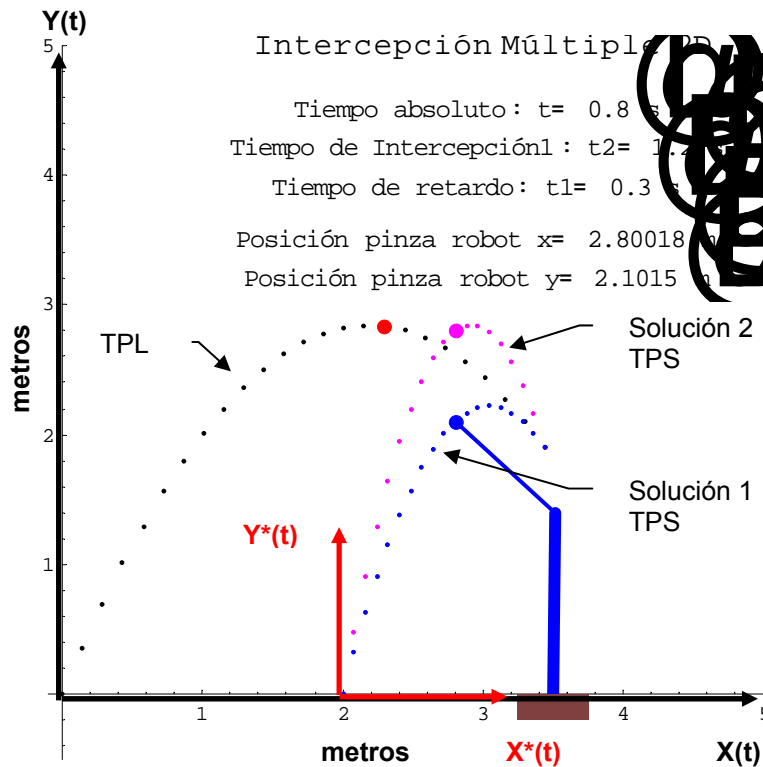
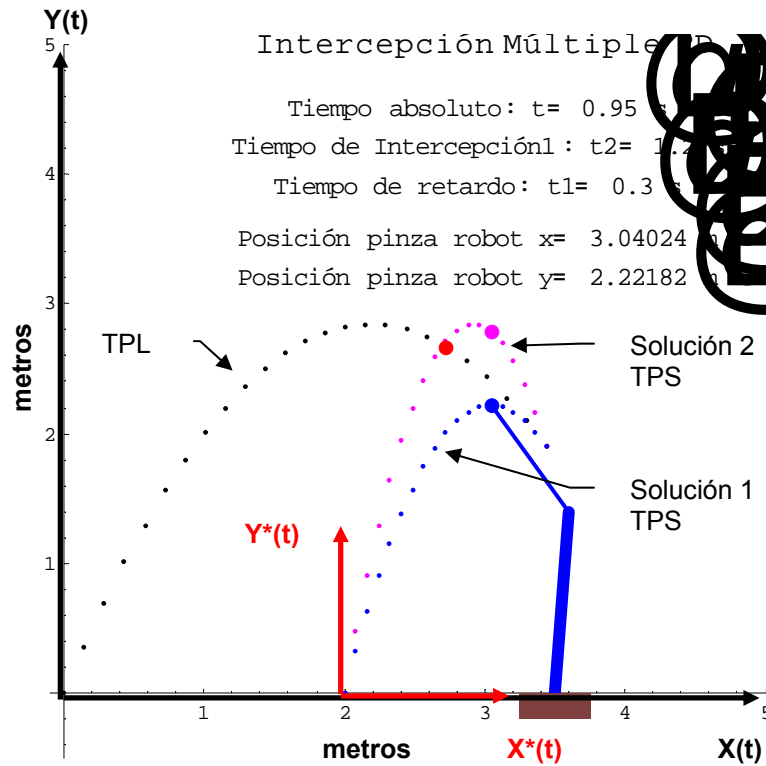


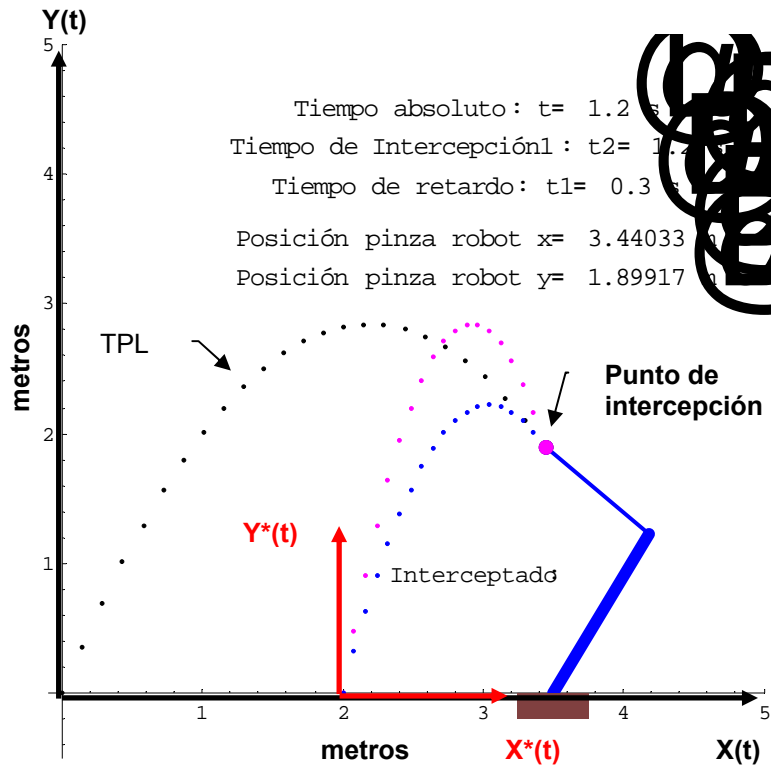
Figura 3.18c Intercepción de la partícula libre usando el caso I para una partícula libre.



**Figura 3.18d** Intercepción de la partícula libre usando el caso I para una partícula libre.

En la figura 3.18d, se puede observar que el efector final del robot interceptor llega a la posición de intercepción antes que la partícula libre TPL. De tal forma, que se adelanta a la posición con la solución 1 TPS.





**Figura 3.18e** Intercepción de la partícula libre usando el caso I para una partícula libre.

En el Anexo A se presenta la secuencia de imágenes completa:

### III.9 INTERCEPCIÓN DE DOS PARTÍCULAS LIBRES VÍA ROBOT INTERCEPTOR DE 2GDL.

Los algoritmos se tomaron del punto II.9, en este caso se interceptan dos partículas libres a una distancia de  $(a,b)$ , unidades desde origen del sistema de coordenadas fijo. Para esta simulación, la coordenada  $b = 0$ . Los algoritmos se presentan a continuación:

$X^*(t) = a, \quad 0 \leq t \leq t_1$ $X^*(t) = a(tt_3b + t_2(c + td + e) + t_2^2f), \quad t > t_1 \quad (3.4)$
---

Donde;

$$a = \frac{1}{(t_1 - t_2)(t_1 - t_3)(t_2 - t_3)}$$

$$b = -at + at_3 + |V_1| \cos \theta_1 t_1 - |V_1| \cos \theta_1 t_1^2$$

$$c = -t_3^2(a + |V_0| \cos \theta_0 (t - t_1))$$

$$d = at - |V_0| \text{Cos}\theta_0 t t_1 + |V_0| \text{Cos}\theta_0 t_1^2$$

$$e = (t^2 - t_1^2) t_3 (|V_0| \text{Cos}\theta_0 - |V_1| \text{Cos}\theta_1)$$

$$f = -at + t_3 (a + |V_1| \text{Cos}\theta_1 (t - t_1))$$

$$Y_1^*(t) = b, \quad 0 \leq t \leq t_1$$

$$Y_1^*(t) = a(t t_3 b + t_2(c + t d + e) + t_2^2 f), \quad t > t_1 \quad (3.5)$$

Donde;

$$a = \frac{1}{(t_1 - t_2)(t_1 - t_3)(t_2 - t_3)}$$

$$b = -bt + bt_3 + |V_1| \text{Sen}\theta_1 t t_1 - |V_1| \text{Sen}\theta_1 t_1^2$$

$$c = -t_3^2 (b + |V_0| \text{Sen}\theta_0 (t - t_1))$$

$$d = bt - |V_0| \text{Sen}\theta_0 t t_1 + |V_0| \text{Sen}\theta_0 t_1^2$$

$$e = (t^2 - t_1^2) t_3 (|V_0| \text{Sen}\theta_0 - |V_1| \text{Sen}\theta_1)$$

$$f = -bt + t_3 (b + |V_1| \text{Sen}\theta_1 (t - t_1))$$

Recordando que:  $Y^*(t) = Y_1^*(t) + Y_2^*(t)$ :

$$Y^*(t) = b, \quad 0 \leq t \leq t_1$$

$$Y^*(t) = Y_1^*(t) + a(t b + t_2 c), \quad t > t_1 \quad (3.6)$$

Donde;

$$a = \frac{1}{2(t_1 - t_2)(t_1 - t_3)}$$

$$b = 2bt + (-2b + g t t_1 - g t_1^2) t_3$$

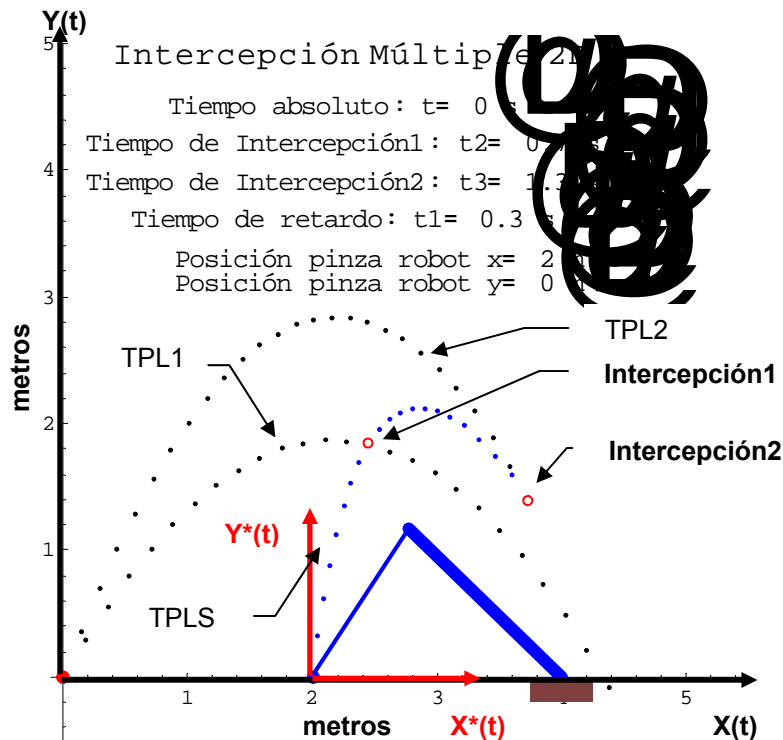
$$c = -2bt + g t^2 t_1 + (2b - g t^2) t_3 + g t_1^2 (-t + t_3)$$

La simulación de la intercepción se realizará con los datos que se presentan a continuación. El listado del programa se encuentra en el Anexo A. (Inte\_par2cb.nb)

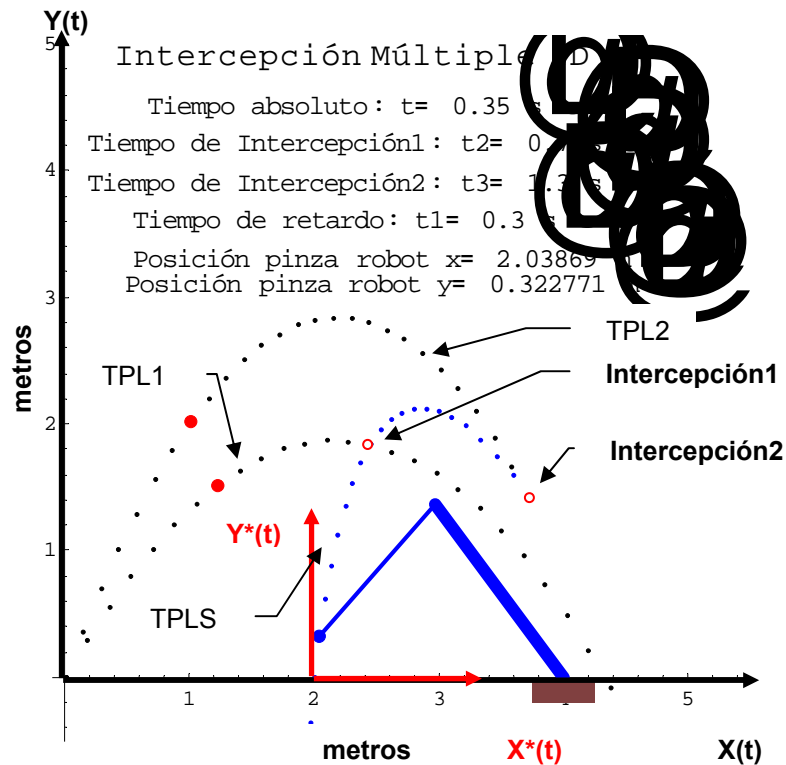
Partícula #1 TPL1		
Velocidad inicial:	$ V_0  =$	7 m/s.
Ángulo de disparo:	$\theta_0 =$	60° grados.
Tiempo de retardo:	$t_1 =$	0.3 seg.
Tiempo de intercepción:	$t_2 =$	0.7 seg.
Cambio de base eje $x^*(t)$	$a =$	2 m
Cambio de base eje $y^*(t)$	$b =$	0 m

Partícula #2 TPL2		
Velocidad inicial:	$ V_0  =$	8 m/s.
Ángulo de disparo:	$\theta_0 =$	69° grados.
Tiempo de retardo:	$t_1 =$	0.3 seg.
Tiempo de intercepción:	$t_2 =$	1.3 seg.
Cambio de base eje $x^*(t)$	$a =$	2 m
Cambio de base eje $y^*(t)$	$b =$	0 m

Con los datos anteriores se pretende ilustrar el caso de intercepción múltiple, los algoritmos se muestran en un programa realizado en el software de cálculo formal **Mathemática® 4.0** en el anexo A. Las trayectorias de las partículas libres están dadas de tal forma que estén dentro del área de trabajo del robot interceptor. Por esa razón difieren de los datos del problema de la página 87.



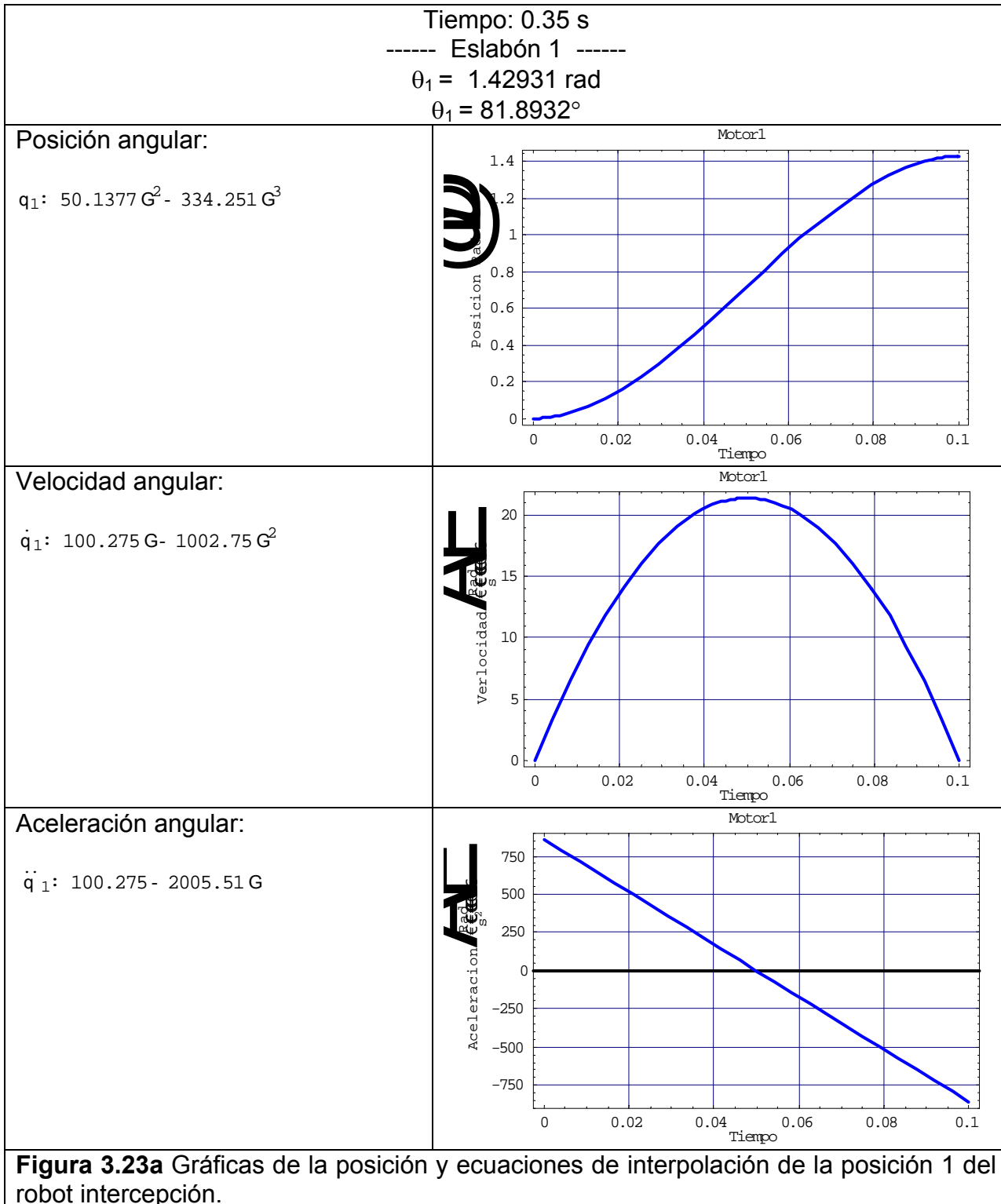
**Figura 3.21** Configuración del robot interceptor en el instante de tiempo absoluto  $t = 0$  s.

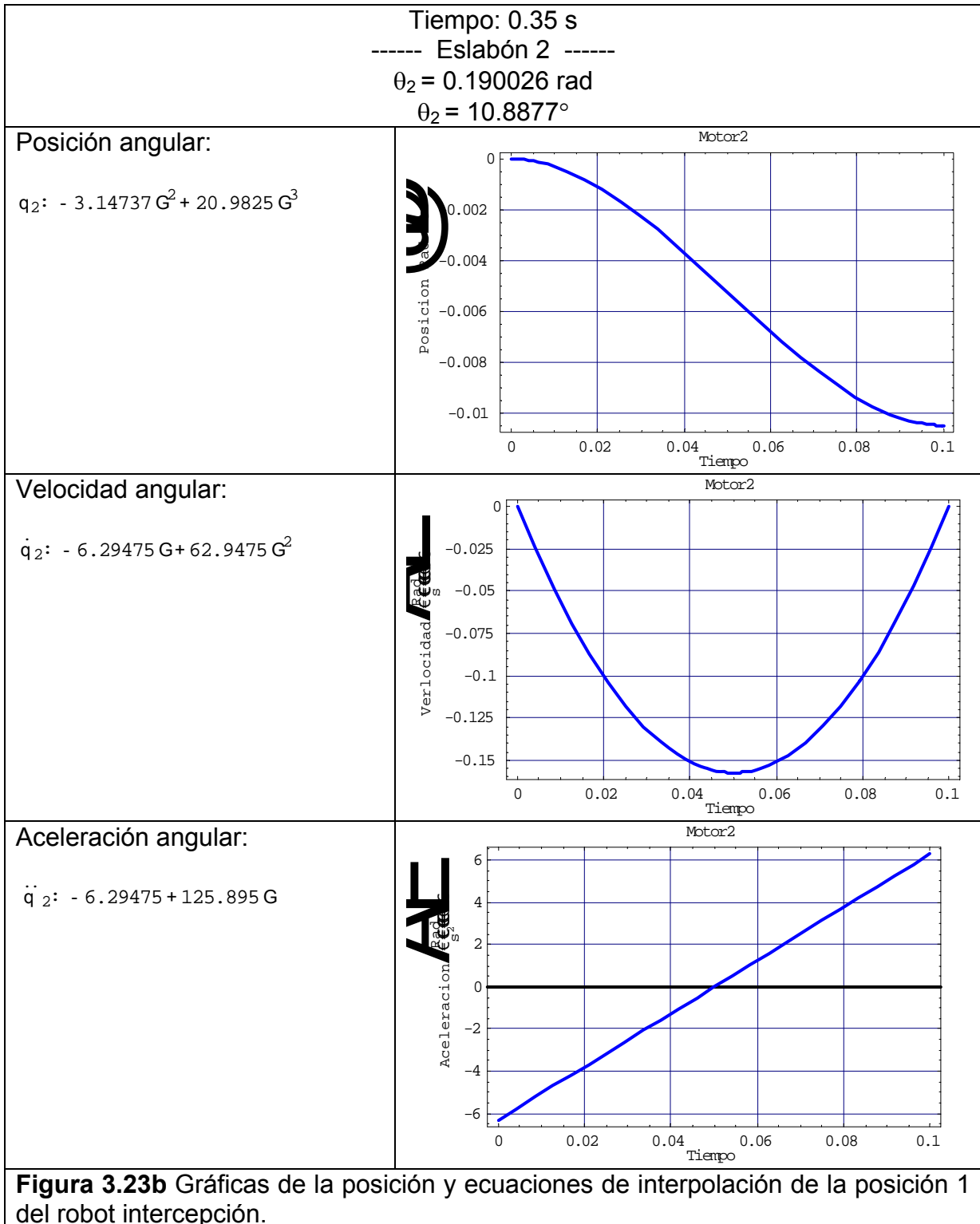


**Figura 3.22** Intercepción para dos partículas libres vía robot interceptor, en el tiempo  $t = 0.35$  s.

En la figura 3.21 se muestra la configuración inicial del robot interceptor preparado para iniciar su movimiento para interceptar las partículas libres. El efector final del robot coincide con el origen del sistema de coordenadas  $(X^*(t), Y^*(t))$  y la base del robot se encuentra en la posición  $(4,0)$  respecto al sistema  $(X(t), Y(t))$ . En la figura 3.22 se muestra la posición del robot interceptor en el tiempo de  $t = 0.35$  s, que es  $0.05$  s después del tiempo de retardo  $t_1$ . El análisis se realizó “discretizando” la trayectoria de la partícula seguidora TPS en intervalos de tiempo de  $0.05$  s, y se calcularon las velocidades y aceleraciones necesarias para satisfacer el tiempo de  $0.05$  s, para cada segmento de la trayectoria hasta completarla. A continuación, se presentarán las ecuaciones obtenidas al aplicar interpolación cúbica y sus respectivas gráficas de la primera posición, ya que las demás posiciones son similares [3.1].

Es importante mencionar, que el robot inició su movimiento en  $t_1 = 0.3$  s, cuando las partículas han estado en movimiento  $0.35$  s, y ya alcanzaron la posición mostrada en la figura 3.22. En las figuras 3.23a y 3.23b, se muestran las ecuaciones de posición, velocidad y aceleración angular de los eslabones 1 y 2 del robot interceptor, en la primera posición que ocurrió a  $0.05$  s, después del tiempo de retardo  $t_1 = 0.3$  s. La forma en que llegan a la posición calculada es del tipo parabólico con velocidad inicial cero y velocidad final cero. De tal manera, que el robot seguirá por tramos, la trayectoria para interceptar las partículas libres. La velocidad angular que tendrán los eslabones del robot será de tipo parabólico y su aceleración constante. Lo anterior, permitirá que el movimiento del robot sea continuo.





En la tabla 3.2 se presentan todas las ecuaciones de posición y velocidad de las articulaciones de los eslabones 1 y 2. El símbolo  $\Gamma$  representa el tiempo total del evento. Las ecuaciones de la trayectoria del efector final de un robot de 2 gdl. Interceptando dos partículas libres con velocidades y ángulos de disparo diferentes.

A continuación se presentan las ecuaciones para cada punto de la trayectoria que generan los algoritmos de intercepción.

Tabla 3.2			
Ecuaciones de movimiento del robot interceptor del evento de la figura 3.22, en términos de coordenadas articulares absolutas			
$\Delta t$	Parámetro cinemático	Eslabón 1	Eslabón 2
0.00	Posición:  Velocidad: Aceleración:	$q_1 = 1.59643 \text{ Rad}$ $q_1 = 91.4688^\circ$ $q_1: 478.93 \text{ G}^2 - 3192.87 \text{ G}^3$ $\dot{q}_1: 957.86 \text{ G} - 9578.6 \text{ G}^2$ $\ddot{q}_1: 957.86 - 19157.2 \text{ G}$	$q_2 = 0.179534 \text{ Rad}$ $q_2 = 10.2866^\circ$ $q_2: 53.8603 \text{ G}^2 - 359.069 \text{ G}^3$ $\dot{q}_2: 107.721 \text{ G} - 1077.21 \text{ G}^2$ $\ddot{q}_2: 107.721 - 2154.41 \text{ G}$
.....	.....	.....	.....
0.30	.....	.....	.....
0.35	Posición:  Velocidad: Aceleración:	$q_1 = 1.42931 \text{ Rad}$ $q_1 = 81.8932^\circ$ $q_1: 50.1377 \text{ G}^2 - 334.251 \text{ G}^3$ $\dot{q}_1: 100.275 \text{ G} - 1002.75 \text{ G}^2$ $\ddot{q}_1: 100.275 - 2005.51 \text{ G}$	$q_2 = 0.190026 \text{ Rad}$ $q_2 = 10.8877^\circ$ $q_2: -3.14737 \text{ G}^2 + 20.9825 \text{ G}^3$ $\dot{q}_2: -6.29475 \text{ G} + 62.9475 \text{ G}^2$ $\ddot{q}_2: -6.29475 + 125.895 \text{ G}$
0.40	Posición:  Velocidad: Aceleración:	$q_1 = 1.28913 \text{ Rad}$ $q_1 = 73.8617^\circ$ $q_1: 42.0533 \text{ G}^2 - 280.355 \text{ G}^3$ $\dot{q}_1: 84.1066 \text{ G} - 841.066 \text{ G}^2$ $\ddot{q}_1: 84.1066 - 1682.13 \text{ G}$	$q_2 = 0.166652 \text{ Rad}$ $q_2 = 9.54845^\circ$ $q_2: 7.01211 \text{ G}^2 - 46.7474 \text{ G}^3$ $\dot{q}_2: 14.0242 \text{ G} - 140.242 \text{ G}^2$ $\ddot{q}_2: 14.0242 - 280.484 \text{ G}$
0.45	Posición:  Velocidad: Aceleración:	$q_1 = 1.17579 \text{ Rad}$ $q_1 = 67.3681^\circ$ $q_1: 34.0003 \text{ G}^2 - 226.669 \text{ G}^3$ $\dot{q}_1: 68.0007 \text{ G} - 680.007 \text{ G}^2$ $\ddot{q}_1: 68.0007 - 1360.01 \text{ G}$	$q_2 = 0.119471 \text{ Rad}$ $q_2 = 6.84518^\circ$ $q_2: 14.1543 \text{ G}^2 - 94.3621 \text{ G}^3$ $\dot{q}_2: 28.3086 \text{ G} - 283.086 \text{ G}^2$ $\ddot{q}_2: 28.3086 - 566.173 \text{ G}$
0.50	Posición:  Velocidad: Aceleración:	$q_1 = 1.08653 \text{ Rad}$ $q_1 = 62.2536^\circ$ $q_1: 26.7791 \text{ G}^2 - 178.527 \text{ G}^3$ $\dot{q}_1: 53.5582 \text{ G} - 535.582 \text{ G}^2$ $\ddot{q}_1: 53.5582 - 1071.16 \text{ G}$	$q_2 = 0.0575351 \text{ Rad}$ $q_2 = 3.29652^\circ$ $q_2: 18.5807 \text{ G}^2 - 123.872 \text{ G}^3$ $\dot{q}_2: 37.1615 \text{ G} - 371.615 \text{ G}^2$ $\ddot{q}_2: 37.1615 - 743.229 \text{ G}$
0.55	Posición:  Velocidad: Aceleración:	$q_1 = 1.01698 \text{ Rad}$ $q_1 = 58.2687^\circ$ $q_1: 20.8649 \text{ G}^2 - 139.099 \text{ G}^3$ $\dot{q}_1: 41.7298 \text{ G} - 417.298 \text{ G}^2$ $\ddot{q}_1: 41.7298 - 834.597 \text{ G}$	$q_2 = -0.0113823 \text{ Rad}$ $q_2 = -0.652157^\circ$ $q_2: 20.6752 \text{ G}^2 - 137.835 \text{ G}^3$ $\dot{q}_2: 41.3504 \text{ G} - 413.504 \text{ G}^2$ $\ddot{q}_2: 41.3504 - 827.008 \text{ G}$

Tabla 3.2 <i>continuación...</i>			
Ecuaciones de movimiento del robot interceptor del evento de la figura 3.22 en términos de coordenadas articulares absolutas			
$\Delta t$		Eslabón 1	Eslabón 2
0.60	Posición:	$q_1 = 0.962241 \text{ Rad}$ $q_1 = 55.1324^\circ$ $q_1: 16.422 \text{ G}^2 - 109.48 \text{ G}^3$	$q_2 = -0.080687 \text{ Rad}$ $q_2 = -4.62303^\circ$ $q_2: 20.7914 \text{ G}^2 - 138.61 \text{ G}^3$
	Velocidad:	$\dot{q}_1: 32.844 \text{ G} - 328.44 \text{ G}^2$	$\dot{q}_2: 41.5829 \text{ G} - 415.829 \text{ G}^2$
	Aceleración:	$\ddot{q}_1: 32.844 - 656.881 \text{ G}$	$\ddot{q}_2: 41.5829 - 831.657 \text{ G}$
0.65	Posición:	$q_1 = 0.9175 \text{ Rad}$ $q_1 = 52.5689^\circ$ $q_1: 13.4224 \text{ G}^2 - 89.4826 \text{ G}^3$	$q_2 = -0.144701 \text{ Rad}$ $q_2 = -8.29075^\circ$ $q_2: 19.2042 \text{ G}^2 - 128.028 \text{ G}^3$
	Velocidad:	$\dot{q}_1: 26.8448 \text{ G} - 268.448 \text{ G}^2$	$\dot{q}_2: 38.4083 \text{ G} - 384.083 \text{ G}^2$
	Aceleración:	$\ddot{q}_1: 26.8448 - 536.896 \text{ G}$	$\ddot{q}_2: 38.4083 - 768.166 \text{ G}$
0.70	Posición:	$q_1 = 0.878345 \text{ Rad}$ $q_1 = 50.3255^\circ$ $q_1: 11.7464 \text{ G}^2 - 78.3093 \text{ G}^3$	$q_2 = -0.198437 \text{ Rad}$ $q_2 = -11.3696^\circ$ $q_2: 16.1209 \text{ G}^2 - 107.473 \text{ G}^3$
	Velocidad:	$\dot{q}_1: 23.4928 \text{ G} - 234.928 \text{ G}^2$	$\dot{q}_2: 32.2418 \text{ G} - 322.418 \text{ G}^2$
	Aceleración:	$\ddot{q}_1: 23.4928 - 469.856 \text{ G}$	$\ddot{q}_2: 32.2418 - 644.835 \text{ G}$
0.75	Posición:	$q_1 = 0.8409 \text{ Rad}$ $q_1 = 48.18^\circ$ $q_1: 11.2336 \text{ G}^2 - 74.8907 \text{ G}^3$	$q_2 = -0.237531 \text{ Rad}$ $q_2 = -13.6095^\circ$ $q_2: 11.7282 \text{ G}^2 - 78.1883 \text{ G}^3$
	Velocidad:	$\dot{q}_1: 22.4672 \text{ G} - 224.672 \text{ G}^2$	$\dot{q}_2: 23.4565 \text{ G} - 234.565 \text{ G}^2$
	Aceleración:	$\ddot{q}_1: 22.4672 - 449.344 \text{ G}$	$\ddot{q}_2: 23.4565 - 469.13 \text{ G}$
0.80	Posición:	$q_1 = 0.801913 \text{ Rad}$ $q_1 = 45.9462^\circ$ $q_1: 11.696 \text{ G}^2 - 77.9734 \text{ G}^3$	$q_2 = -0.25836 \text{ Rad}$ $q_2 = -14.8029^\circ$ $q_2: 6.24849 \text{ G}^2 - 41.6566 \text{ G}^3$
	Velocidad:	$\dot{q}_1: 23.392 \text{ G} - 233.92 \text{ G}^2$	$\dot{q}_2: 12.497 \text{ G} - 124.97 \text{ G}^2$
	Aceleración:	$\ddot{q}_1: 23.392 - 467.841 \text{ G}$	$\ddot{q}_2: 12.497 - 249.94 \text{ G}$
0.85	Posición:	$q_1 = 0.758853 \text{ Rad}$ $q_1 = 43.4791^\circ$ $q_1: 12.9179 \text{ G}^2 - 86.1195 \text{ G}^3$	$q_2 = -0.258288 \text{ Rad}$ $q_2 = -14.7988^\circ$ $q_2: -0.0214447 \text{ G}^2 + 0.142964 \text{ G}^3$
	Velocidad:	$\dot{q}_1: 25.8358 \text{ G} - 258.358 \text{ G}^2$	$\dot{q}_2: -0.0428893 \text{ G} + 0.428893 \text{ G}^2$
	Aceleración:	$\ddot{q}_1: 25.8358 - 516.717 \text{ G}$	$\ddot{q}_2: -0.0428893 + 0.857786 \text{ G}$
0.90	Posición:	$q_1 = 0.709974 \text{ Rad}$ $q_1 = 40.6785^\circ$ $q_1: 14.6639 \text{ G}^2 - 97.7593 \text{ G}^3$	$q_2 = -0.235913 \text{ Rad}$ $q_2 = -13.5168^\circ$ $q_2: -6.71253 \text{ G}^2 + 44.7502 \text{ G}^3$
	Velocidad:	$\dot{q}_1: 29.3278 \text{ G} - 293.278 \text{ G}^2$	$\dot{q}_2: -13.4251 \text{ G} + 134.251 \text{ G}^2$
	Aceleración:	$\ddot{q}_1: 29.3278 - 586.556 \text{ G}$	$\ddot{q}_2: -13.4251 + 268.501 \text{ G}$
0.95	Posición:	$q_1 = 0.654291 \text{ Rad}$ $q_1 = 37.4881^\circ$ $q_1: 16.7048 \text{ G}^2 - 111.365 \text{ G}^3$	$q_2 = -0.191121 \text{ Rad}$ $q_2 = -10.9504^\circ$ $q_2: -13.4377 \text{ G}^2 + 89.5849 \text{ G}^3$
	Velocidad:	$\dot{q}_1: 33.4096 \text{ G} - 334.096 \text{ G}^2$	$\dot{q}_2: -26.8755 \text{ G} + 268.755 \text{ G}^2$
	Aceleración:	$\ddot{q}_1: 33.4096 - 668.192 \text{ G}$	$\ddot{q}_2: -26.8755 + 537.509 \text{ G}$



1.00	Posición:	$q_1 = 0.591452 \text{ Rad}$ $q_1 = 33.8877^\circ$ $q_1: 18.8516 G^2 - 125.678 G^3$	$q_2 = -0.124903 \text{ Rad}$ $q_2 = -7.15641^\circ$ $q_2: -19.8653 G^2 + 132.435 G^3$
	Velocidad:	$\dot{q}_1: 37.7033 G - 377.033 G^2$	$\dot{q}_2: -39.7306 G + 397.306 G^2$
	Aceleración:	$\ddot{q}_1: 37.7033 - 754.066 G$	$\ddot{q}_2: -39.7306 + 794.612 G$

Tabla 3.2 <i>continuación...</i>			
Ecuaciones de movimiento del robot interceptor del evento de la figura 3.22, en términos de coordenadas articulares absolutas			
$\Delta t$		Eslabón 1	Eslabón 2
1.05	Posición:	$q_1 = 0.521513 \text{ Rad}$ $q_1 = 29.8805^\circ$ $q_1: 20.9817 G^2 - 139.878 G^3$	$q_2 = -0.0389828 \text{ Rad}$ $q_2 = -2.23355^\circ$ $q_2: -25.776 G^2 + 171.84 G^3$
	Velocidad:	$\dot{q}_1: 41.9633 G - 419.633 G^2$	$\dot{q}_2: -51.552 G + 515.52 G^2$
	Aceleración:	$\ddot{q}_1: 41.9633 - 839.266 G$	$\ddot{q}_2: -51.552 + 1031.04 G$
1.10	Posición:	$q_1 = 0.444694 \text{ Rad}$ $q_1 = 25.4791^\circ$ $q_1: 23.0457 G^2 - 153.638 G^3$	$q_2 = 0.0646004 \text{ Rad}$ $q_2 = 3.70133^\circ$ $q_2: -31.075 G^2 + 207.166 G^3$
	Velocidad:	$\dot{q}_1: 46.0914 G - 460.914 G^2$	$\dot{q}_2: -62.1499 G + 621.499 G^2$
	Aceleración:	$\ddot{q}_1: 46.0914 - 921.828 G$	$\ddot{q}_2: -62.1499 + 1243. G$
1.15	Posición:	$q_1 = 0.361154 \text{ Rad}$ $q_1 = 20.6926^\circ$ $q_1: 25.0621 G^2 - 167.081 G^3$	$q_2 = 0.183822 \text{ Rad}$ $q_2 = 10.5322^\circ$ $q_2: -35.7664 G^2 + 238.443 G^3$
	Velocidad:	$\dot{q}_1: 50.1242 G - 501.242 G^2$	$\dot{q}_2: -71.5328 G + 715.328 G^2$
	Aceleración:	$\ddot{q}_1: 50.1242 - 1002.48 G$	$\ddot{q}_2: -71.5328 + 1430.66 G$
1.20	Posición:	$q_1 = 0.270785 \text{ Rad}$ $q_1 = 15.5149^\circ$ $q_1: 27.1106 G^2 - 180.737 G^3$	$q_2 = 0.316865 \text{ Rad}$ $q_2 = 18.155^\circ$ $q_2: -39.913 G^2 + 266.087 G^3$
	Velocidad:	$\dot{q}_1: 54.2212 G - 542.212 G^2$	$\dot{q}_2: -79.826 G + 798.26 G^2$
	Aceleración:	$\ddot{q}_1: 54.2212 - 1084.42 G$	$\ddot{q}_2: -79.826 + 1596.52 G$
1.25	Posición:	$q_1 = 0.172977 \text{ Rad}$ $q_1 = 9.91086^\circ$ $q_1: 29.3425 G^2 - 195.616 G^3$	$q_2 = 0.462186 \text{ Rad}$ $q_2 = 26.4813^\circ$ $q_2: -43.5962 G^2 + 290.642 G^3$
	Velocidad:	$\dot{q}_1: 58.6849 G - 586.849 G^2$	$\dot{q}_2: -87.1925 G + 871.925 G^2$
	Aceleración:	$\ddot{q}_1: 58.6849 - 1173.7 G$	$\ddot{q}_2: -87.1925 + 1743.85 G$
1.30	Posición:	$q_1 = 0.0662016 \text{ Rad}$ $q_1 = 3.79307^\circ$ $q_1: 32.0327 G^2 - 213.551 G^3$	$q_2 = 0.618443 \text{ Rad}$ $q_2 = 35.4342^\circ$ $q_2: -46.8772 G^2 + 312.514 G^3$
	Velocidad:	$\dot{q}_1: 64.0654 G - 640.654 G^2$	$\dot{q}_2: -93.7543 G + 937.543 G^2$
	Aceleración:	$\ddot{q}_1: 64.0654 - 1281.31 G$	$\ddot{q}_2: -93.7543 + 1875.09 G$

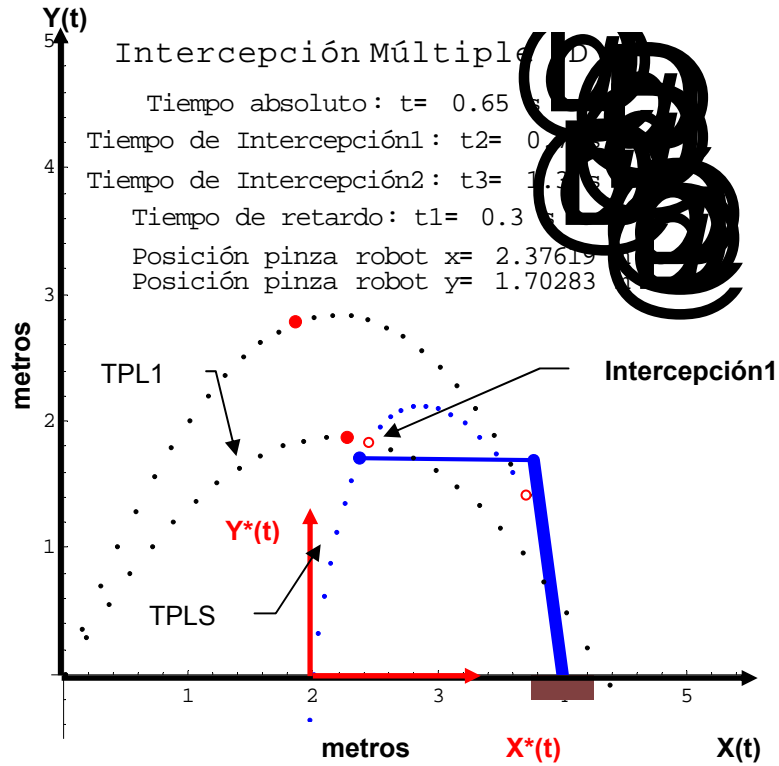


Figura 3.24a Intercepción para dos partículas libres vía robot interceptor, en el tiempo  $t = 0.65$  s.

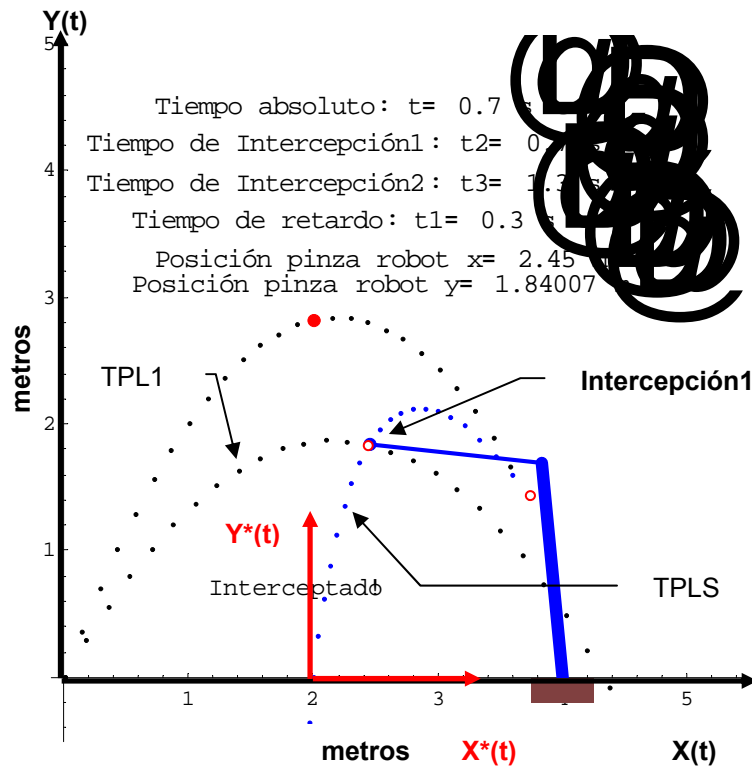
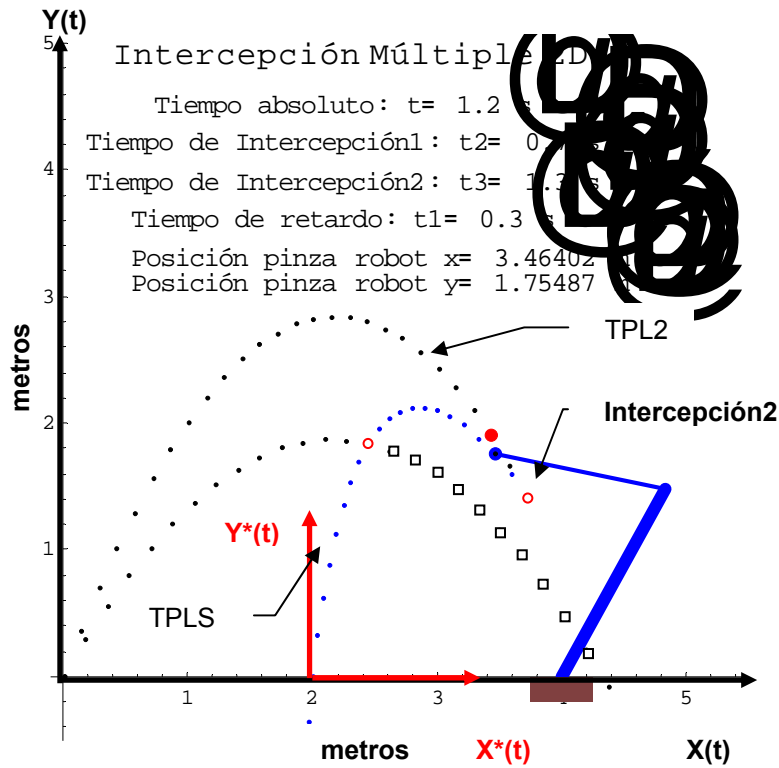
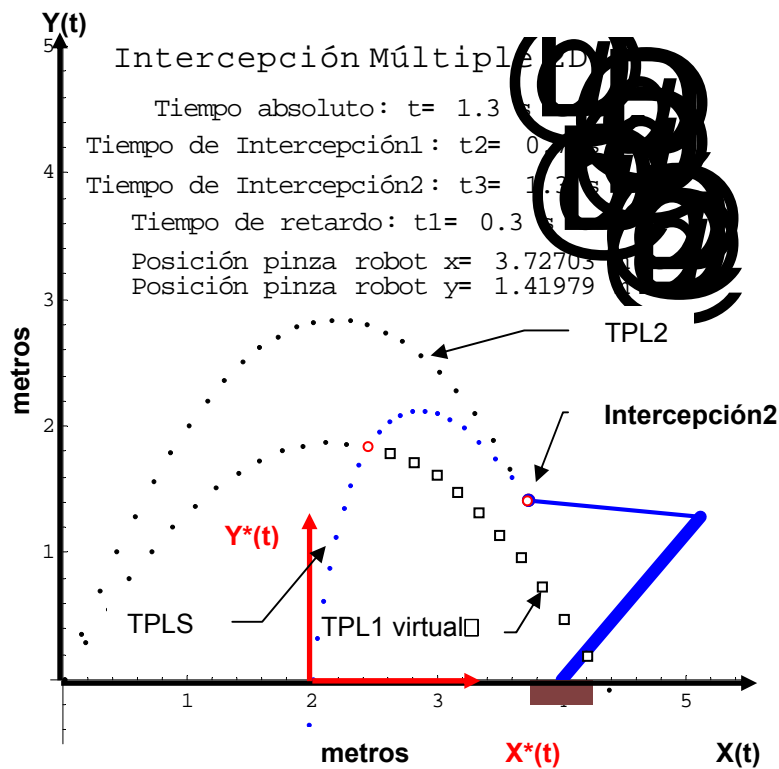


Figura 3.24b Intercepción para dos partículas libres vía robot interceptor en el tiempo  $t = 0.70$  s.



**Figura 3.24c** Intercepción para dos partículas libres vía robot interceptor en el tiempo  $t = 1.20$  s.

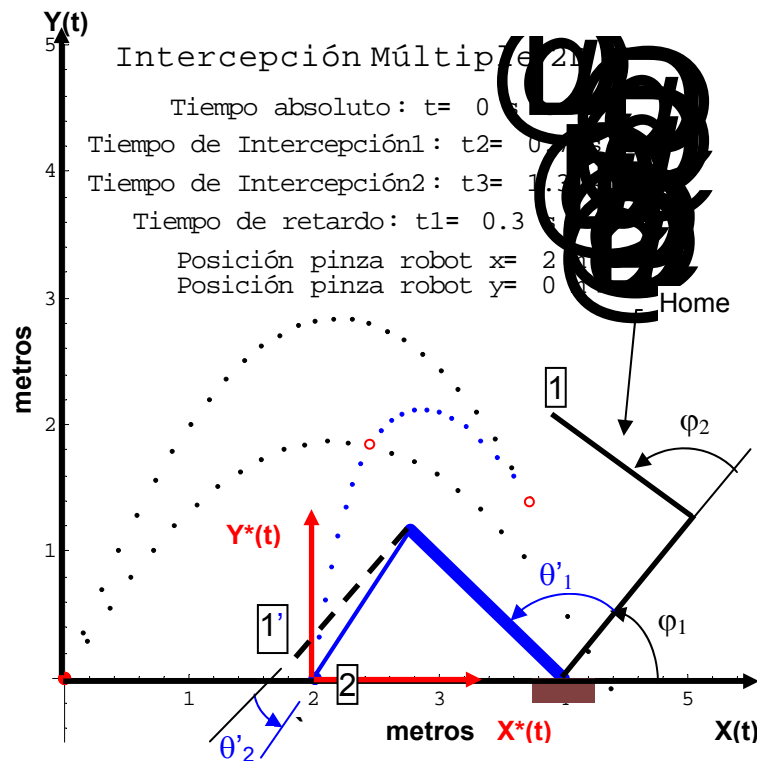


**Figura 3.24d** Intercepción para dos partículas libres vía robot interceptor en el tiempo  $t = 0.30$  s.

En las figuras 3.24a a la 3.24d se aprecia la secuencia simplificada de la intercepción realizada por el robot interceptor. En la figura 3.24b se puede observar la primera partícula libre TPL1 interceptada. La trayectoria que se muestra en el simulador en la TPL1 es virtual, ya no existe, porque la partícula TPL1 ha sido interceptada y sólo se muestra con fines ilustrativos. Mientras esto ocurre, en la simulación la otra partícula libre TPL2 continúa su movimiento a la posición de intercepción 2 y el robot no se detiene hasta que intercepta la segunda partícula, ver figura 3.24d.

### III.10 ANÁLISIS DE LAS COORDENADAS RELATIVAS DEL ROBOT INTERCEPTOR.

En relación con el análisis de las coordenadas relativas del robot interceptor para el evento de la figura 3.21, ahora se presentarán sólo las gráficas de los puntos de intercepción 1 y el punto de intercepción 2. Está vez se obtendrán los valores relativos a la última posición de los eslabones 1 y 2. Cabe mencionar, que las rotaciones en sentido antihorario se consideran positivas. Las coordenadas relativas son necesarias porque algunos controladores se programan utilizando coordenadas relativas.

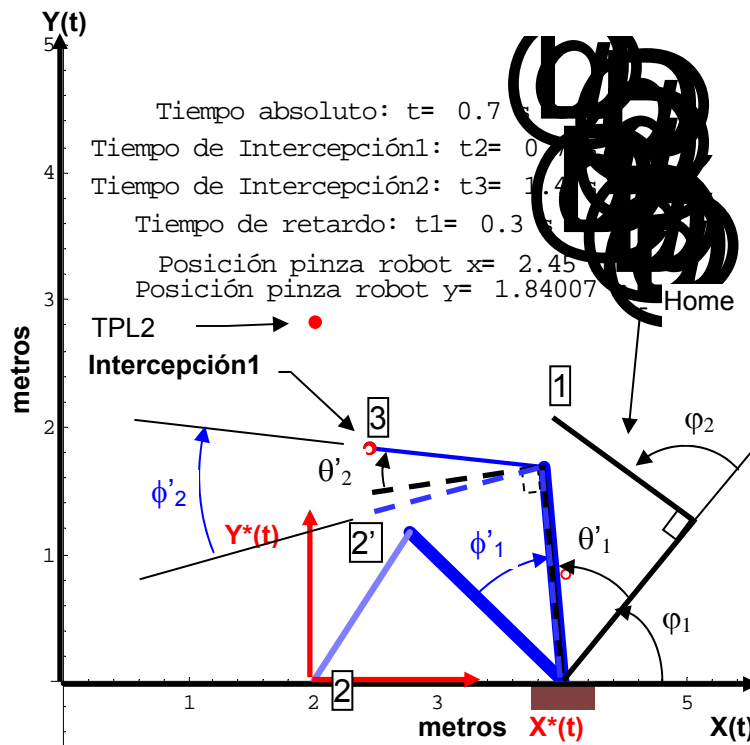


**Figura 3.24e** Configuración de inicio de intercepción del robot interceptor en el tiempo absoluto  $t = 0$  s.

En la figura 3.24e, el tiempo absoluto es  $t = 0$  s, la posición del robot es la configuración de inicio de intercepción, ver figura 3.13. Aquí no importa el tiempo de llegada del robot interceptor a esa posición, ya que es una posición de preparación para que el robot intercepte las partículas. Sin embargo, a través de la solución de la cinemática inversa se encontraron los siguientes valores para la posición 2. La posición 1 (Home), tiene los siguientes ángulos  $\phi_1 = 45^\circ$  y  $\phi_2 = 90^\circ$ , con respecto al eje X. Ahora bien, para alcanzar

la posición 2, primero rotó el eslabón 1,  $\theta'_1 = 91.4688^\circ$ , y el eslabón 2 fue arrastrado ese desplazamiento angular. Luego el eslabón 2 sólo rotó  $\theta'_2 = 10.2866^\circ$ . Con estos valores se calcularán las interpolaciones cúbicas para implementar posteriormente el controlador PID.

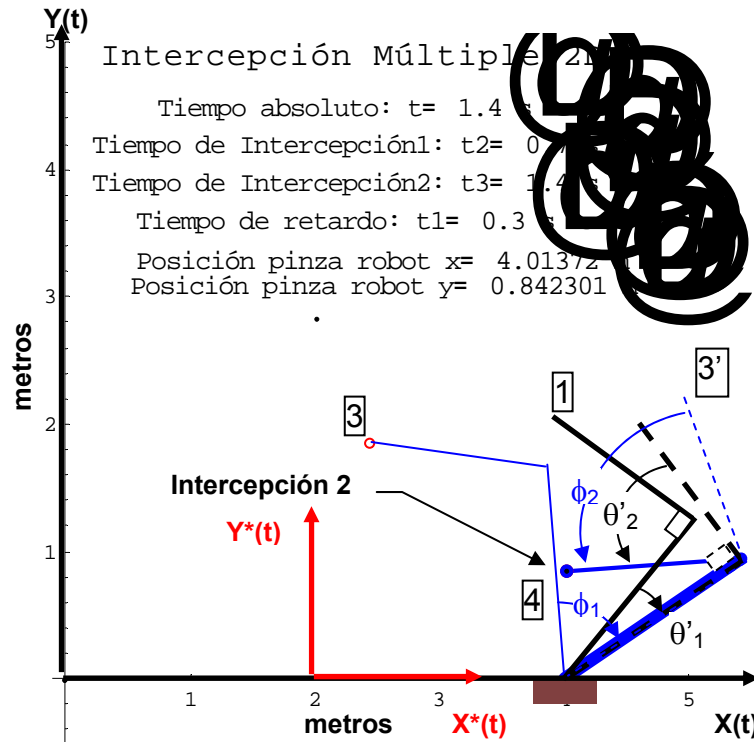
En la figura 3.24f, se presenta el primer movimiento que realiza el robot interceptor para alcanzar la posición de Intercepción 1. Es decir, captura la primera partícula libre TPL1. En el tiempo de intercepción  $t_2 = 0.7$ , predefinido.



**Figura 3.24f** Intercepción 1 del robot en el tiempo absoluto  $t = 0.7$  s.

Se puede observar en la figura 3.24f, que se necesita encontrar los desplazamientos angulares desde la posición 2, para llegar a la posición 3. Aquí, es importante mencionar que los valores de la cinemática inversa están referidos a la configuración de Home, que es la posición 1. Ahora, el eslabón 1 rotó  $\theta'_1 = 50.3255^\circ$  llevando a la posición 2' al eslabón 2, luego el eslabón 2 rotó  $\theta'_2 = -11.3696^\circ$  llegando a la posición 3, pero desde la posición de Home. Sin embargo, se necesitan obtener los ángulos, pero desde la posición 2. Por lo tanto, El eslabón 1 rota un ángulo  $\phi'_1 = -41^\circ$  y el eslabón 2 queda en la posición 2', luego el eslabón 2 rota  $\phi'_2 = -23^\circ$ , que queda en la posición 3.

Ahora se presentará la intercepción 2, en tiempo absoluto  $t = 1.3$  s, y de igual forma se obtendrán los ángulos relativos de los desplazamientos angulares de los eslabones del robot interceptor, ver figura 3.24g.



**Figura 3.24g** Intercepción 2 del robot interceptor en el tiempo absoluto  $t = 1.4$  s.

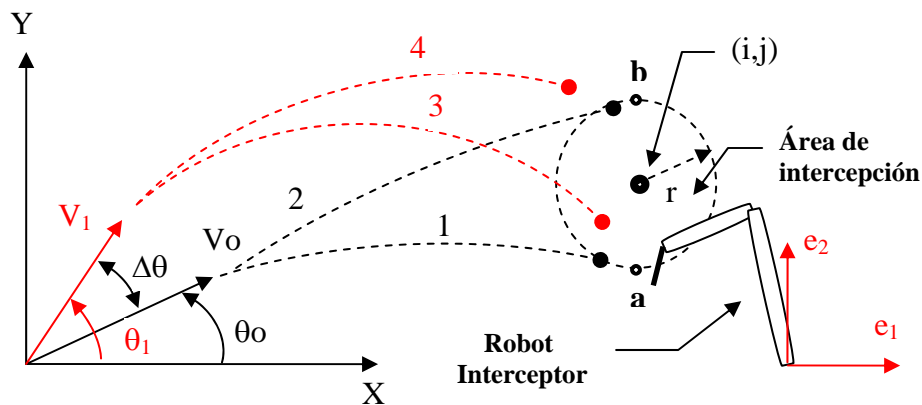
**Tabla 3.2a**  
Ecuaciones en coordenadas articulares relativas:  
Del evento mostrado en las figuras 3.24e, 3.24f, 3.24g.

$\Delta t$	Parámetro cinemático	Eslabón 1	Eslabón 2
0.00	Home	$\theta_1 = 45^\circ$	$\theta_2 = 90^\circ$
0.00	Posición:  Velocidad: Aceleración:	$q_1 = 2.38183 \text{ Rad}$ $q_1 = 136.469^\circ$ $q_1: 0. + 14.5826 G^2 - 13.8882 G^3$ $\dot{q}_1: 29.1653 G - 41.6647 G^2$ $\ddot{q}_1: 29.1653 - 83.3293 G$	$q_2 = 1.75033 \text{ Rad}$ $q_2 = 100.287^\circ$ $q_2: 0. + 10.7163 G^2 - 10.206 G^3$ $\dot{q}_2: 21.4326 G - 30.618 G^2$ $\ddot{q}_2: 21.4326 - 61.2361 G$
0.70	Posición:  Velocidad: Aceleración:	$q_1 = 1.66624 \text{ Rad}$ $q_1 = 95.4688^\circ$ $q_1: 0. + 10.2015 G^2 - 9.71571 G^3$ $\dot{q}_1: 20.403 G - 29.1471 G^2$ $\ddot{q}_1: 20.403 - 58.2943 G$	$q_2 = 1.34891 \text{ Rad}$ $q_2 = 77.2866^\circ$ $q_2: 0. + 8.25861 G^2 - 7.86534 G^3$ $\dot{q}_2: 16.5172 G - 23.596 G^2$ $\ddot{q}_2: 16.5172 - 47.192 G$
1.40	Posición:  Velocidad: Aceleración:	$q_1 = 0.584141 \text{ Rad}$ $q_1 = 33.4688^\circ$ $q_1: 0. + 3.57637 G^2 - 3.40607 G^3$ $\dot{q}_1: 7.15274 G - 10.2182 G^2$ $\ddot{q}_1: 7.15274 - 20.4364 G$	$q_2 = 2.60554 \text{ Rad}$ $q_2 = 149.287^\circ$ $q_2: 0. + 15.9523 G^2 - 15.1927 G^3$ $\dot{q}_2: 31.9046 G - 45.578 G^2$ $\ddot{q}_2: 31.9046 - 91.156 G$

En la figura 3.24g, de la cinemática inversa el eslabón 1 rota  $\theta'_1 = -11.0109^\circ$ , con respecto a Home posición 1, y luego el eslabón 2 gira  $\theta'_2 = 60.4377^\circ$  y llega a la posición 4. Nuevamente, se necesitan los desplazamientos angulares desde la posición relativa, que es en este caso la posición 3. Para ello, el eslabón 1 rota un ángulo  $\phi_1 = -62^\circ$  quedando en la posición 3', luego el eslabón 2 tendría que rotar  $\phi_2 = 72^\circ$  para alcanzar la posición 4. Las líneas punteadas representan las posiciones intermedias.

### III.11 DEFINICIÓN DEL ÁREA DE TRABAJO.

En este tema se define el rango de operación del robot interceptor, el cual está definido en función de las trayectorias que describen las partículas libres, en este caso; TPL1 y TPL2. Las trayectorias de las partículas libres 1 y 2, respectivamente.



**Figura 3.25** Esquema que ilustra el rango de intercepción del robot interceptor.

Debido a que el robot interceptor tiene sus límites de velocidad y de área de trabajo, es necesario que se establezcan los valores mínimos y máximos de los parámetros cinemáticos de las partículas libres: TPL1 y TPL2. Es decir, cuales son los rangos de velocidad y ángulos de disparo de las partículas entre si, para que el robot interceptor capture las partículas libres dentro de su área de trabajo. Así como también, la velocidad relativa que debe existir entre las partículas. En la figura 3.25, se establece una circunferencia de radio  $r$ , y centro ubicado en  $(i,j)$  con respecto al sistema de coordenadas inercial  $X,Y$ . Los puntos  $a(x,y)$  y  $b(x,y)$  están definidos en dos puntos que pertenecen a la circunferencia a lo largo de una línea vertical, como se muestra en la figura 3.25. Las ecuaciones que describen la trayectoria de las partículas TPL1 y TPL2 son:

$$\begin{aligned} X_0(t) &= |V_0| \cos \theta_0 t \\ Y_0(t) &= |V_0| \sin \theta_0 t - \frac{1}{2}gt^2 \end{aligned} \tag{3.7a}$$

$$\begin{aligned} X_1(t) &= |V_1| \cos \theta_1 t \\ Y_1(t) &= |V_1| \sin \theta_1 t - \frac{1}{2}gt^2 \end{aligned} \tag{3.7b}$$

Ahora se determina la ecuación que define los ángulos mínimo y máximo para una velocidad dada de la trayectoria de la partícula libre 1, TPL1. Por lo tanto, de la ecuación (3.6). Cabe señalar que en este cálculo se sustituye  $X(t)$  y  $Y(t)$  por  $X$  y  $Y$ , respectivamente.

$$X = (|V_0| \cos \alpha)t \quad (3.8)$$

$$t = \frac{X}{|V_0| \cos \alpha} \quad (3.9)$$

$$Y = (|V_0| \sin \alpha)t - \frac{1}{2}gt^2 \quad (3.10)$$

Sustituyendo la ecuación (3.9), en la ecuación (3.10), simplificando y utilizando las siguientes entidades trigonométricas:

$$\frac{1}{\cos^2 \alpha} = \sec^2 \alpha \quad \text{y} \quad \sec^2 \alpha = 1 + \tan^2 \alpha$$

Se obtiene el siguiente modelo:

$$\left(1 + \frac{2|V_0|^2 Y}{gX^2}\right) - \frac{2|V_0|^2}{gX} \tan \alpha + \tan^2 \alpha \quad (3.11)$$

Al resolver la ecuación (3.11), se obtienen dos soluciones, que son los ángulos de disparo de la partícula para una velocidad inicial  $|V_0|$  dada. La velocidad de las partículas será igual, debido a que los algoritmos serán aplicados para capturar objetos sobre una banda transportadora que se mueve a velocidad constante. Aquí, el punto **a** y **b** tiene las coordenadas  $(X, Y)$ , es el punto a alcanzar, y el ángulo  $\alpha$ , puede tomar los valores  $\theta_0$  y  $\theta_1$  [3.31].

La información que se debe plantear para calcular el rango de intercepción es la siguiente:

La velocidad inicial de la partícula 1 TPL1, la posición del punto a y b, cuyas coordenadas son  $(X, Y)$ .

Es importante mencionar, que el área de intercepción debe estar contenida dentro del área de trabajo del robot interceptor.



---

## III.12 ESQUEMA DE CONTROL IMPLEMENTADO APLICADO AL ROBOT INTERCEPTOR PARA LA TAREA DE INTERCEPCIÓN.

### III.12.1 EL MODELO DINÁMICO DEL ROBOT INTERCEPTOR

Antes de aplicar algún esquema de control al robot interceptor es necesario obtener su modelo dinámico. La robótica es un campo complejo que envuelve varias disciplinas, tales como la física, las propiedades de los materiales, la estática y la dinámica, la electrónica, la teoría de control, visión, procesamiento de imágenes, programación de computadoras y manufactura [3.3]. Uno de los aspectos más complicados de la robótica es la obtención del modelo dinámico de un robot manipulador que tiene por objetivo conocer la relación entre el movimiento del robot y las fuerzas implicadas en el mismo. El modelo dinámico relaciona:

- 1.- La localización del efector final del robot definida por sus variables articulares o por las coordenadas de localización de su extremo, por sus derivadas: velocidad y aceleración.
- 2.- Las fuerzas y pares aplicados en las articulaciones (o en el extremo del robot).
- 3.- Los parámetros dimensionales del robot como son: longitud, masas e inercias de sus elementos.

Obtener el modelo dinámico es uno de los aspectos más complejos de la robótica. Además, es más factible encontrar su solución numérica que la analítica ya que el sistema de ecuaciones es de segundo grado, altamente no lineal y multivariable. Se debe integrar el sistema de ecuaciones en forma iterativa mediante algún método numérico: Runge-Kutta, etc. [3.4,3.11].

Hoy en día, la temática del control de robots manipuladores ha atraído la atención de una parte de la comunidad científica de control automático. Uno de los principales puntos de interés en los robots se ubica en el diseño de sistemas controladores de seguimiento para realizar movimientos de alta velocidad y exactitud. Generalmente, los controladores de seguimiento se basan en el modelo dinámico del robot manipulador, por lo que es de vital importancia contar con un modelo que describa fielmente los efectos dinámicos de su comportamiento. Sin embargo, no es suficiente contar con la estructura completa del modelo dinámico, ya que subsisten problemas de naturaleza práctica como son la obtención numérica de los parámetros dinámicos del robot manipulador (centros de masa, masas y momentos de inercia), lo que generalmente se convierte en una tarea difícil, más aún cuando es posible desarmar el robot manipulador [3.12]. Es decir, el modelo es imprescindible para conseguir los siguientes fines:

- 1.- Simulación del movimiento del robot.
- 2.- Diseño y evaluación de la estructura mecánica del robot.
- 3.- Dimensionamiento de los actuadores.
- 4.- Diseño y evaluación del control dinámico del robot.

El último fin es de gran importancia, ya que la calidad del control dinámico depende de la precisión y velocidad de sus movimientos. Por ese motivo, se han creado metodologías para obtener el modelo dinámico de robots con  $n$  grados de libertad. Se estudiarán las siguientes formulaciones: que se aplican a robots con eslabones rígidos, en forma sistemática para obtener el modelo dinámico.

- 1.- Euler-Lagrange (E-L). Método energético.
- 2.- Newton- Euler(N-E). Método de Fuerzas y Aceleraciones.

### III.12.2 CARACTERÍSTICAS DE LAS FORMULACIONES

1.- Euler-Lagrange: Es fácil y sistemático, pero calcular los coeficientes dinámicos  $D_{ik}$ ,  $h_{ikm}$  y  $C_i$  requieren de muchas operaciones aritméticas. Por lo tanto, es muy difícil utilizarlo en el control en tiempo real a menos que se simplifiquen las ecuaciones.

2.- Newton-Euler: Es una alternativa para obtener las ecuaciones eficientemente, pero laboriosa e involucra operaciones de producto cruz. Se utilizan ecuaciones recursivas hacia adelante y hacia atrás, y se van aplicando secuencialmente a los eslabones. La propagación recursiva hacia adelante obtiene información cinemática: velocidades lineales y angulares, aceleraciones lineales y angulares del centro de masa de los eslabones, desde un sistema de referencia fijo. La propagación recursiva hacia atrás obtiene los momentos y las fuerzas ejercidas sobre cada eslabón, desde el efector final hasta la base del robot, y además se puede lograr en forma simple el control lineal articular en tiempo real. La eficiencia de las ecuaciones de E-L parte de la matriz de coordenadas homogéneas  $4 \times 4$ . Mientras que la eficiencia de N-E se basa en formulación vectorial y su naturaleza recursiva, pero las ecuaciones recursivas destruyen la estructura del modelo dinámico para el diseño de control en el espacio de estados [3.5].

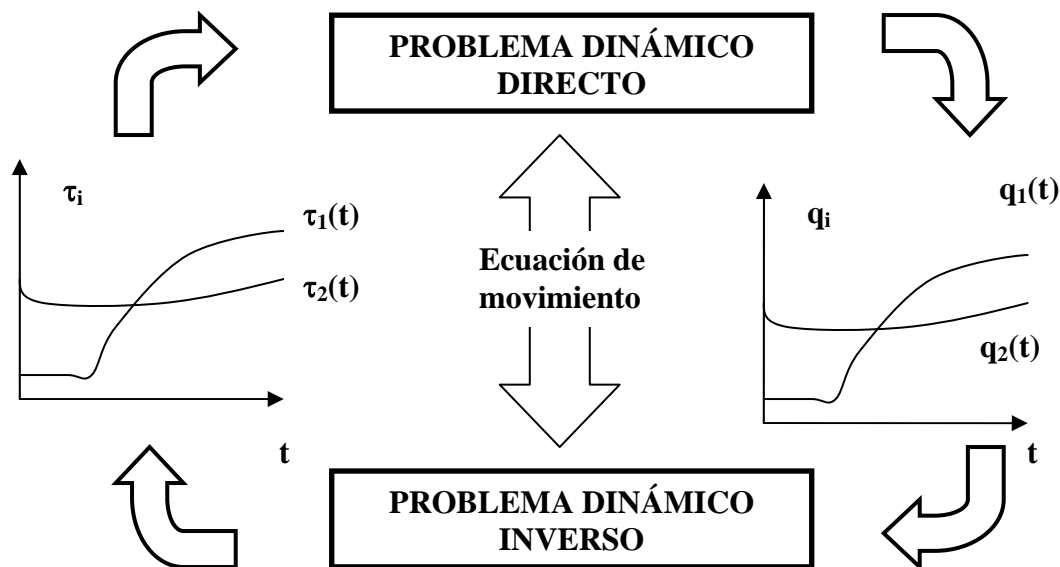


Figura 3.26 Esquema que representa los dos problemas dinámicos.

Se puede encontrar con mucha frecuencia en la literatura científica de robot manipuladores que el primer método es el más utilizado para desarrollar el modelo dinámico de robots. Además, se debe contar con la estructura del modelo dinámico para implantar alguna técnica de control en nuestro proyecto. Por lo tanto, se utilizará la formulación de Euler-Lagrange para obtener el modelo dinámico del robot interceptor.

En la figura 3.26, se muestra el modelo dinámico directo: que expresa la evolución temporal de las coordenadas articulares del robot en función de las fuerzas y pares que intervienen en ellas, y el modelo dinámico Inverso: que expresa las fuerzas y pares que intervienen en función de la evolución de las coordenadas articulares y sus derivadas [3.4]. A continuación se presenta la formulación de Euler-Lagrange para obtener el modelo dinámico del robot interceptor. La formulación que se presenta es para el caso de robots de  $n$  grados de libertad con juntas rotacionales o prismáticas.

Ecuación de movimiento simplificada:

$$\tau_i = \mathbf{D}(q(t))\ddot{q} + \mathbf{h}(q(t), \dot{q}(t)) + \mathbf{C}(q(t)) \quad (3.12)$$

donde  $i = 1, 2, 3, \dots, n$

$$\tau_i(t) = (\tau_1(t), \tau_2(t), \tau_3(t), \dots, \tau_n(t))^T \quad (3.13)$$

$n \times 1$  vector de fuerzas y torques generalizados :

$$q(t) = (q_1(t), q_2(t), q_3(t), \dots, q_n(t))^T \quad (3.14)$$

$n \times 1$  vector de variables articulares.

$$\dot{q}(t) = (\dot{q}_1(t), \dot{q}_2(t), \dot{q}_3(t), \dots, \dot{q}_n(t))^T \quad (3.15)$$

$n \times 1$  vector de velocidades articulares

$$\ddot{q}(t) = (\ddot{q}_1(t), \ddot{q}_2(t), \ddot{q}_3(t), \dots, \ddot{q}_n(t))^T \quad (3.16)$$

$n \times 1$  vector de aceleraciones articulares

$\mathbf{D}(q(t))$ ,  $n \times n$  matriz simétrica inercial de masas.

$$D_{ik} = \sum_{j=\max(i,k)}^n \text{Tr}(\mathbf{U}_j \mathbf{J}_j^T \mathbf{U}_j), \quad i, k = 1, 2, 3, \dots, n \quad (3.17)$$

$\mathbf{h}(q(t), \dot{q}(t))$ ,  $n \times 1$  vector no lineal de fuerzas de Coriolis y centrifugas.

$$h(q(t), \dot{q}(t)) = (h_1, h_2, h_3, \dots, h_n)^T \quad (3.18)$$

$$h_i = \sum_{k=1}^n \sum_{m=1}^n h_{ikm} \dot{q}_k \dot{q}_m \quad (3.19)$$

$$h_{ikm} = \sum_{j=\max(i,k,m)}^n \text{Tr}(\mathbf{U}_j \mathbf{J}_{km}^T \mathbf{U}_j), \quad i, k, m = 1, 2, 3, \dots, n \quad (3.20)$$

$C(q(t))$   $n \times 1$  vector de fuerzas por gravedad.

$$C(q(t)) = (C_1, C_2, C_3, \dots, C_n)^T \quad (3.21)$$

$$C_i = \sum_{j=1}^n (-m_j g U_{ji}^j \bar{r}_j), \quad i = 1, 2, 3, \dots, n \quad (3.22)$$

Considérese un robot de  $n$  grados de libertad, formado por eslabones rígidos conectados por articulaciones libres de elasticidad (rígidas). La energía total  $E$  del robot manipulador está dada por la suma de la energía cinética  $K$  más la energía potencial  $U$ :

$$E(q, \dot{q}) = K(q, \dot{q}) + U(q), \quad (3.23)$$

donde  $q, \dot{q} \in \mathcal{R}^n$  representan vectores de posición y velocidad articular, respectivamente.

El lagrangiano  $\ell(q, \dot{q})$  de un robot manipulador de  $n$  grados de libertad es definido como la diferencia entre la energía cinética  $K$  y la energía potencial  $U$ :

$$\ell(q, \dot{q}) = K(q, \dot{q}) - U(q) \quad (3.24)$$

Aquí se considera que la energía potencial  $U$  es debida a fuerzas conservativas tales como la fuerza de gravedad. Las ecuaciones de movimiento de Euler-Lagrange de un robot manipulador de  $n$  grado de libertad están dadas por:

$$\frac{d}{dt} \left[ \frac{\partial \ell(q, \dot{q})}{\partial \dot{q}} \right] - \frac{\partial \ell(q, \dot{q})}{\partial q} = \tau - f(\dot{q}, \tau) \quad (3.25)$$

donde  $\tau \in \mathcal{R}^n$  es el vector de fuerzas y pares generalizados que actúan sobre las articulaciones del robot y  $f(\dot{q}, \tau) \in \mathcal{R}^n$  es el vector de fuerzas o pares de fricción presente en las articulaciones del mismo. La energía cinética  $K(q, \dot{q})$  asociada a un robot manipulador es una función cuadrática del vector de velocidad  $\dot{q}$  que puede expresarse como:

$$K(q, \dot{q}) = \frac{1}{2} \dot{q}^T M(q) \dot{q} \quad (3.26)$$

donde  $M(q) \in \mathcal{R}^{n \times n}$  es la Matriz de inercia del manipulador, la cual tiene las propiedades de ser simétrica y definida positiva. La energía potencial  $U(q)$  tiene una dependencia del vector de posiciones  $q$ , pero en general no tiene una forma específica como el caso de la energía cinética  $K(q, \dot{q})$ .

El lagrangiano  $\ell(q, \dot{q})$ , dado por la ecuación (3.24), toma la siguiente forma:

$$\ell(q, \dot{q}) = \frac{1}{2} \dot{q}^T M(q) \dot{q} - U(q) \quad (3.27)$$

con esta forma del lagrangiano, las ecuaciones de movimiento de Euler-Lagrange pueden escribirse en forma compacta como:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = \tau - f(\dot{q}, \tau) \quad (3.28)$$

donde  $C(q, \dot{q}) \in \mathbb{R}^{n \times n}$  es la matriz de fuerzas centrífugas y de Coriolis,  $g(q) \in \mathbb{R}^n$  es el vector de fuerzas o pares gravitacionales obtenido como el gradiente de la energía  $U(q)$  debido a la acción de la gravedad. La fuerza de fricción  $f(\dot{q}, \tau)$  se considera como energía disipativa en velocidades diferentes a cero y con entradas acotadas dentro del primer y tercer cuadrante, lo que permite considerar los modelos tradicionales de fricción viscosa y de Coulomb **[3.13]**. En velocidades cero únicamente está presente la fricción estática satisfaciendo la ecuación (3.25).

$$f(0, \tau_i) = \tau_i - g_i(q) \quad (3.29)$$

para  $-\bar{f}_i \leq \tau_i - g_i(q) \leq \bar{f}_i$  donde  $\bar{f}_i$  es el límite del par de fricción estática para la pésima articulación **[3.14]**. En términos del vector de estado  $[q^T, \dot{q}^T]^T$ , el modelo dinámico del robot manipulador (3.28), puede ser escrito de la siguiente forma:

$$\frac{d}{dt} \begin{bmatrix} q \\ \dot{q} \end{bmatrix} = \begin{bmatrix} \dot{q} \\ M^{-1}(q)[\tau - f(\dot{q}, \tau) - C(q, \dot{q})\dot{q} - g(q)] \end{bmatrix} \quad (3.30)$$

Es importante mencionar que el desarrollo para la obtención del modelo dinámico del robot interceptor se ha tomado de **[3.6, ..., 3.10]**.

Utilizando las ecuaciones (3.17) a la (3.22), se desarrolló el modelo dinámico del robot interceptor de 2 gdl. Este se muestra en la ecuación (3.25), que se obtuvo mediante el programa mostrado en **[3.15]**.

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{bmatrix} l_2^2 m_2 + 2l_1 l_2 m_2 c_2 + l_1^2 (m_1 + m_2) & l_2^2 m_2 + l_1 l_2 m_2 c_2 \\ l_2^2 m_2 + l_1 l_2 m_2 c_2 & l_2^2 m_2 \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} m_2 l_2 g c_{12} + (m_1 + m_2) l_1 g c_1 \\ m_2 l_2 g c_{12} \end{bmatrix} + \begin{bmatrix} -m_2 l_1 l_2 s_2 \dot{\theta}_2^2 - 2m_2 l_1 l_2 s_2 \theta_1 \dot{\theta}_2 \\ m_2 l_1 l_2 s_2 \dot{\theta}_1^2 \end{bmatrix} \quad (3.31)$$

El vector de fricciones se ha ignorado debido a que las velocidades angulares de los eslabones serán altas para la tarea a desarrollar, ya que en este caso la fricción se ignora por ser pequeña comparada con las fuerzas inerciales, centrífugas y de Coriolis del robot interceptor **[3.7]**.

Relacionando las coordenadas articulares  $q$ , con las coordenadas del efector final del robot interceptor  $X_r$  dadas por:

$$\begin{aligned} X_r &= P(q) \\ \dot{X}_r &= J(q)\dot{q} \\ \ddot{X}_r &= \dot{J}(q)\dot{q} + J(q)\ddot{q} \end{aligned} \quad (3.32)$$

donde  $P(q)$  representa la cinemática directa del efector final respecto a las coordenadas articulares, y  $J(q)$  es la matriz del Jacobiano.

Sustituyendo la ecuación (3.32) en la ecuación (3.23), se ha ignorando el término de fricción, se obtiene:

$$MJ^{-1}\{\dot{X}_r - \dot{J}J^{-1}\dot{X}_r\} + CJ^{-1}\dot{X}_r + g = \tau \quad (3.33)$$

Reordenando los términos, se obtiene el modelo dinámico del robot interceptor como:

$$MJ^{-1}\ddot{X}_r + \{C - MJ^{-1}\dot{J}\}J^{-1}\dot{X}_r + g = \tau \quad (3.34)$$

en la ecuación (3.34), el vector de torques  $\tau$  está sujeto a la siguiente restricción.

$$|\tau_i| \leq |\tau_{i\text{máx}}|, \quad i=1,2,3,\dots,n \quad (3.35)$$

donde  $\tau_{i\text{máx}}$  es el máximo torque que puede entregar el  $i$ -ésimo servo actuador. La relación entre la aceleración  $\ddot{X}_r$  y el torque  $\tau$  necesario para producir dicha aceleración, es lineal [3.30]. Aunque en este trabajo se desarrolló el control en coordenadas articulares debido a la naturaleza de la tarea del robot interceptor y al tiempo en el que se deben ejecutar los cálculos en tiempo real vía computadora.

### III.13 CONTROL DE POSICIÓN PURA EN COORDENADAS ARTICULARES

La mayoría de las tareas que un robot manipulador realiza requieren de la ejecución de una trayectoria predeterminada. Al problema de controlar esa trayectoria se le denomina control de movimiento. El tipo de movimiento que realiza el robot manipulador puede involucrar la interacción física con el medio ambiente; en tal caso se le denomina movimiento restringido, en cual puede caer en alguna de las siguientes clasificaciones: control de fuerza, control de impedancia, control híbrido fuerza/impedancia, etc.

Aplicaciones típicas para movimiento restringido son: pulido de superficies y desbastado de materiales, etc. Por otro lado, cuando el robot se desplaza libremente en su espacio de trabajo, sin que el extremo final entre en contacto con su entorno físico, se habla de movimiento no restringido o movimiento libre. Aplicaciones tales como transportación y pintado de objetos son ejemplos de movimiento no restringido.

El problema de control de movimiento (no restringido) de robots manipuladores consiste en determinar los pares aplicados por los actuadores a sus articulaciones, de tal forma que las posiciones asociadas a las coordenadas articulares del robot sigan con exactitud una posición deseada que varía con en el tiempo. En términos más formales, el objetivo de control de movimiento consiste en determinar  $\tau$  de tal forma que el error de posición articular tienda a cero en forma asintótica.

Como caso particular del control de movimiento se encuentra el denominado control de posición pura o regulación, el cual consiste en especificar en el espacio de trabajo del robot manipulador un punto deseado (consiste en el tiempo). La problemática de control de posición pura consiste en llevar el extremo final del robot a la posición deseada independiente de su posición inicial. En la mayoría de las aplicaciones industriales, los

---

robots manipuladores son controlados a través de controladores proporcional-derivativo, los cuales son ampliamente utilizados por su efectividad para regulación [3.16].

Sin embargo, estos controladores no toman en cuenta el hecho de que los servo actuadores están diseñados para entregar cierta capacidad limitada de par, de ahí que se requiere una cuidadosa sintonización en las ganancias del controlador para no saturar los servo actuadores y por consiguiente evitar el deterioro en el desempeño del controlador. La desventaja anterior del controlador PD, ha motivado el interés teórico y práctico para diseñar controladores de posición pura con pares acotados. El problema de control para mantener la acción de control acotada dentro de ciertos límites predeterminados, se le denomina regulación con acciones de control acotadas.

Un ejemplo de controlador con acciones de control acotadas es el de Saturación Proporcional Derivativo propuesto en [3.17], el cual emplea una función no lineal acotada del error de posición y de la velocidad para enfrentar el problema de control de posición pura con pares acotados y que ha demostrado experimentalmente tener un desempeño muy superior al controlador tradicional PD [3.18]. Debido a que el robot experimental se mueve en un plano vertical, todos los controladores incluyen compensación de gravedad.

### III.14 CONTROL DE MOVIMIENTO EN COORDENADAS ARTICULARES

Hoy en día, el interés de los robots manipuladores se ubica en su facultad de realizar movimientos de alta velocidad y con alto grado de precisión. El control de movimiento o control de trayectorias es uno de los tópicos más importantes en el control de robots manipuladores. El problema del control de movimiento de robots manipuladores consiste en encontrar una función vectorial de pares aplicados a sus articulaciones, de tal forma que las posiciones asociadas a las coordenadas articuladas al robot sigan con exactitud una posición deseada en el tiempo.

Los controladores con base en el modelo del robot son particularmente empleados en control de movimiento, los cuales son usados cuando el robot se mueve libremente en su espacio de trabajo siguiendo una trayectoria deseada sin interactuar con su medio ambiente. El problema de control de movimiento ha recibido fuertemente la atención en la comunidad de robótica y como resultado se ha reportado en la literatura diversos controladores. Entre los que han mostrado tener mejor desempeño se encuentran: control de par calculado, control PD+ [3.19], PD con precompensación calculada y PD con compensación [3.20].

Los controladores anteriores se basan en el modelo dinámico del robot, debido a que incluyen la dinámica completa del robot manipulador en la estructura del controlador. La exactitud, desempeño y robustez de esos controladores dependen del grado de precisión con que sean conocidos los parámetros dinámicos que describen el modelo. El control de movimiento de robots manipuladores ha sido ampliamente estudiado en simulaciones. Sin embargo, la evaluación experimental de controladores basados en el modelo del robot manipulador es un problema de origen práctico, el cual ha quedado evidenciado en la literatura científica del área. Lo anterior se debe a la falta de robots

experimentales adecuados, así como la dificultad que presenta conocer el valor nominal de los parámetros dinámicos del robot manipulador.

No obstante, el número de evaluaciones experimentales con robots manipuladores ha ido en aumento en los últimos diez años. Por ejemplo [3.21-3.26] entre otros han presentado evaluaciones experimentales del controlador Par-Calculado y del controlador PD, dando resultados satisfactorios. Por lo tanto, se utilizará en este trabajo la técnica de control de Par-Calculado para el seguimiento de la trayectoria del efector final del robot interceptor.

### III.15 CONTROL BASADO EN EL MODELO PAR-CALCULADO

El control denominado Par-Calculado fue una de las primeras estructuras de control de movimiento basadas en el modelo dinámico del manipulador. Es decir, que hace uso explícito del conocimiento de  $M(q), C(q, \dot{q})$  y  $g(q)$ , la ley de control está dada por:

$$\tau = M(q)\ddot{q}_d + K_v\dot{\tilde{q}} + K_p\tilde{q} + C(q, \dot{q})\dot{q} + g(q) \tag{3.36}$$

donde  $K_v$  Y  $K_p$  son matrices simétricas positivas. La información sobre la trayectoria de movimiento deseada  $q_d(t)$ ,  $\dot{q}_d(t)$  y  $\ddot{q}_d(t)$ , así como la medición de  $q(t)$  y  $\dot{q}(t)$  son empleadas en la realización de la ecuación (3.36), donde  $\tilde{q}(t) = q_d(t) - q$ ,  $\dot{\tilde{q}}(t) = \dot{q}_d(t) - \dot{q}$ , ver figura 3.27.

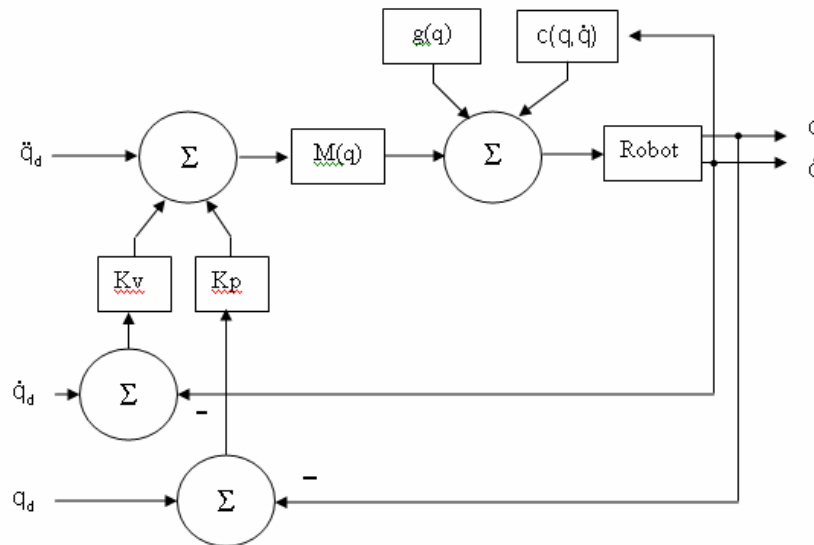


Figura 3.27 Esquema de Control Par-Calculado.

La ecuación de malla cerrada se obtiene reemplazando la acción de control  $\tau$  de la ecuación (3.36) en la ecuación del modelo del robot ecuación (3.28).

$$M(q)\ddot{q} = M(q)\ddot{q}_d + K_v\dot{\tilde{q}} + K_p\tilde{q} + C(q, \dot{q})\dot{q} + g(q) \tag{3.37}$$



y debido a que  $M(q)$  es invertible, la ecuación anterior se reduce a:

$$\ddot{q}_d + K_v \dot{\tilde{q}} + K_p \tilde{q} = 0 \quad (3.38)$$

Es decir, bajo la suposición de que existe una cancelación exacta del modelo dinámico que interviene en el lazo de retroalimentación, entonces el control Par-calculado realiza el desacoplamiento dinámico de todas las articulaciones del robot, resultando el siguiente sistema lineal, ver ecuación (3.38), o en forma equivalente:

$$\begin{aligned} \frac{d}{dt} \begin{bmatrix} \tilde{q} \\ \dot{\tilde{q}} \end{bmatrix} &= \begin{bmatrix} \dot{\tilde{q}} \\ -K_p \tilde{q} - K_v \dot{\tilde{q}} \end{bmatrix} \\ \frac{d}{dt} \begin{bmatrix} \tilde{q} \\ \dot{\tilde{q}} \end{bmatrix} &= \begin{bmatrix} 0 & I \\ -K_p & -K_v \end{bmatrix} \begin{bmatrix} \tilde{q} \\ \dot{\tilde{q}} \end{bmatrix} \end{aligned} \quad (3.39)$$

donde  $I \in \mathbb{R}^{n \times n}$  es la matriz identidad de dimensión  $n$ .

Es importante observar que la ecuación de malla cerrada representa una ecuación diferencial lineal autónoma cuyo único estado de equilibrio es  $\begin{bmatrix} \tilde{q} \\ \dot{\tilde{q}} \end{bmatrix} = 0 \in \mathbb{R}^{2n}$ , la unicidad de equilibrio se obtiene demostrando que:

$$\det \begin{bmatrix} 0 & I \\ -K_p & -K_v \end{bmatrix} \neq 0 \quad (3.40)$$

lo cual es inmediato el hecho de que  $\det(K_p) > 0$  y que:

$$\begin{aligned} \det \begin{bmatrix} 0 & I \\ -K_p & -K_v \end{bmatrix} &= \det(-K_v) \det(-K_v^{-1} K_p) \\ &= (\det(-K_v)) (-\det(-K_v^{-1} K_p)) \\ &= \det(K_p) \end{aligned}$$

debido a que la matriz  $K_v = K_v^T$  se selecciona definida positiva, entonces existe un escalar real positivo  $\varepsilon^*$  tal que para todo  $\varepsilon \in (0, \varepsilon^*]$ :

$$K_v - \varepsilon I > 0 \quad (3.41)$$

Específicamente, cualquier  $\varepsilon^* < \lambda_{\min}(K_v)$  es adecuado. Considerado la desigualdad (3.35) y la positividad de  $K_p$ , éstas implican a su vez que:

$$K_p + \varepsilon K_v - \varepsilon^2 I > 0 \quad (3.42)$$

Procediendo ahora al análisis de estabilidad del equilibrio (origen) de la ecuación de malla cerrada (3.34), considérese la siguiente función candidata de Lyapunov:

$$V(\tilde{q}, \dot{\tilde{q}}) = \frac{1}{2} \begin{pmatrix} \tilde{q} \\ \dot{\tilde{q}} \end{pmatrix}^T \begin{pmatrix} K_p + \varepsilon K_v & \varepsilon I \\ \varepsilon I & I \end{pmatrix} \begin{pmatrix} \tilde{q} \\ \dot{\tilde{q}} \end{pmatrix}$$

$$= \frac{1}{2}(\dot{\tilde{q}} + \varepsilon\tilde{q})^T(\dot{\tilde{q}} + \varepsilon\tilde{q}) + \frac{1}{2}\tilde{q}^T(K_P + \varepsilon K_V - \varepsilon^2 I)\tilde{q} \geq 0 \quad (3.43)$$

donde el escalar  $\varepsilon > 0$  satisface (3.41). La expresión en la ecuación (3.34) es equivalente a:

$$V(\tilde{q}, \dot{\tilde{q}}) = \frac{1}{2}\dot{\tilde{q}}^T\dot{\tilde{q}} + \frac{1}{2}\tilde{q}^T(K_P + \varepsilon K_V)\tilde{q} + \varepsilon\tilde{q}^T\dot{\tilde{q}} \quad (3.44)$$

cuya derivada con respecto al tiempo es:

$$\dot{V}(\tilde{q}, \dot{\tilde{q}}) = \ddot{\tilde{q}}^T\dot{\tilde{q}} + \tilde{q}^T(K_P + \varepsilon K_V)\dot{\tilde{q}} + \varepsilon\dot{\tilde{q}}^T\dot{\tilde{q}} + \varepsilon\dot{\tilde{q}}^T\ddot{\tilde{q}} \quad (3.45)$$

sustituyendo  $\ddot{\tilde{q}}$  de la ecuación de malla cerrada (3.34) en la expresión anterior y realizando simplificaciones se obtiene:

$$\dot{V}(\tilde{q}, \dot{\tilde{q}}) = -\dot{\tilde{q}}^T(K_V - \varepsilon I)\dot{\tilde{q}} - \varepsilon\dot{\tilde{q}}^T K_P \tilde{q} \quad (3.46)$$

$$= -\begin{pmatrix} \tilde{q} \\ \dot{\tilde{q}} \end{pmatrix}^T \begin{pmatrix} \varepsilon K_P & 0 \\ 0 & K_V - \varepsilon I \end{pmatrix} \begin{pmatrix} \tilde{q} \\ \dot{\tilde{q}} \end{pmatrix} \leq 0 \quad (3.47)$$

como  $K_V - \varepsilon I > 0$  (3.41) y  $K_P > 0$ , la función  $-\dot{V}$  (3.44) es una función definida positiva. Por lo tanto, el origen  $\begin{bmatrix} \tilde{q} \\ \dot{\tilde{q}} \end{bmatrix} = 0$  es global y asintóticamente estable [3.27] y entonces:

$$\lim_{t \rightarrow \infty} \tilde{q}(t) = 0$$

$$\lim_{t \rightarrow \infty} \dot{\tilde{q}}(t) = 0$$

esto es, se verifica el objetivo de control de movimiento.

Para fines prácticos, la elección de las matrices  $K_P$  Y  $K_V$  pueden obtenerse utilizando la ecuación (3.44):

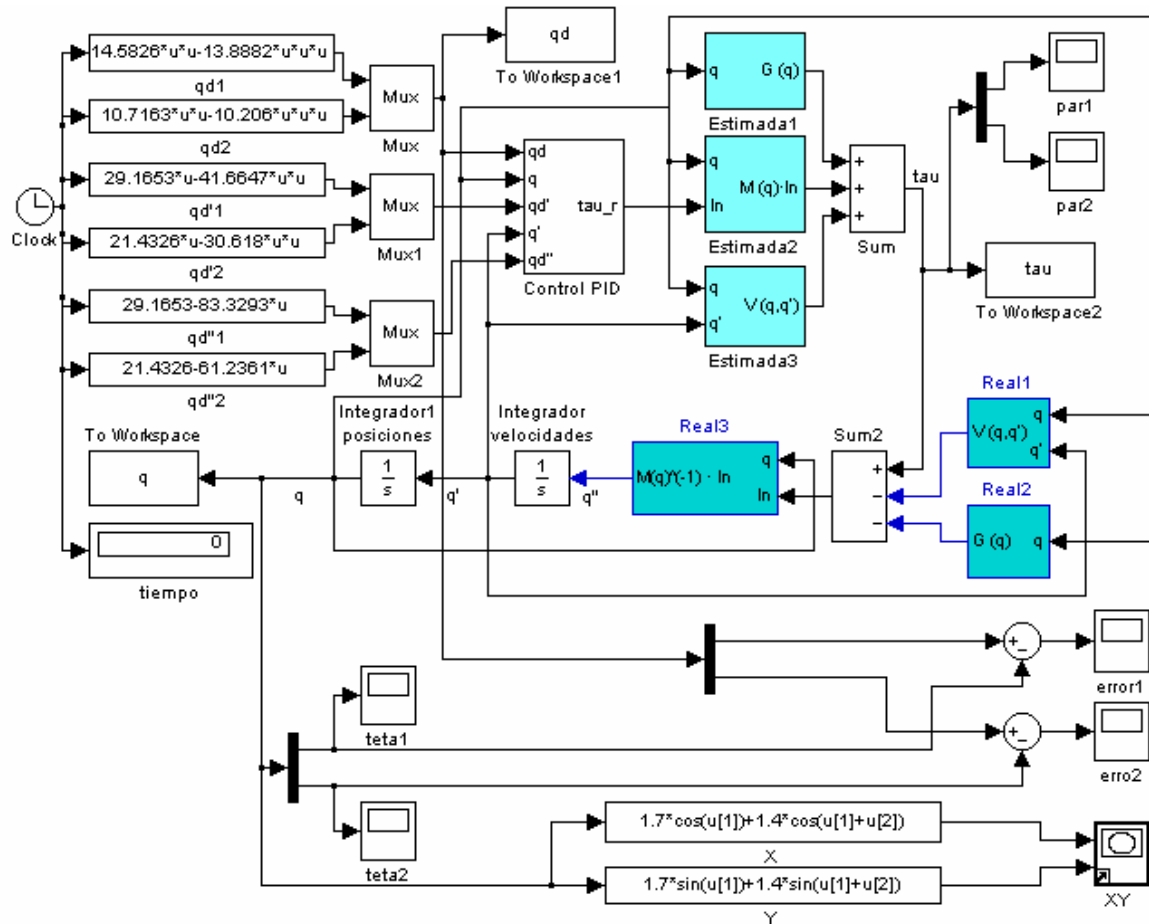
$$K_P = \text{diag}\{w_1^2, \dots, w_n^2\}$$

$$K_V = \text{diag}\{2w_1, \dots, 2w_n\} \quad (3.48)$$

Esta elección resulta en un sistema de malla cerrada desacoplado, cada unión responde igual que un sistema lineal de segundo orden críticamente amortiguado con ancho de banda  $w_i$ . El cual determina la velocidad de respuesta de la unión y en consecuencia la tasa de decaimiento exponencial de los errores  $\tilde{q}(t)$  y  $\dot{\tilde{q}}(t)$  [3.27,3.28,3.29].

**III.16 SIMULACIÓN DEL CONTROL BASADO EN EL MODELO PAR-CALCULADO IMPLEMENTADO AL ROBOT INTERCEPTOR**

A continuación se presenta el esquema de control en la figura 3.28 [3.28], que se aplicará el esquema de control para alcanzar las posiciones de intercepción de la tabla 3.2a, que son coordenadas relativas. Además, se presentarán las gráficas de posición y velocidad del efector final. En el modelo presentado se introducen las funciones cúbicas de interpolación calculadas anteriormente. Se seleccionaron las ganancias que generaron un desempeño satisfactorio. Es importante mencionar que el sistema de coordenadas se fija en la base del robot, para efectos de la simulación. Sin embargo, se sabe que se localiza a 4 unidades sobre el eje de las X(t) y a 0 unidades en el eje Y(t) del sistema inercial fijo.



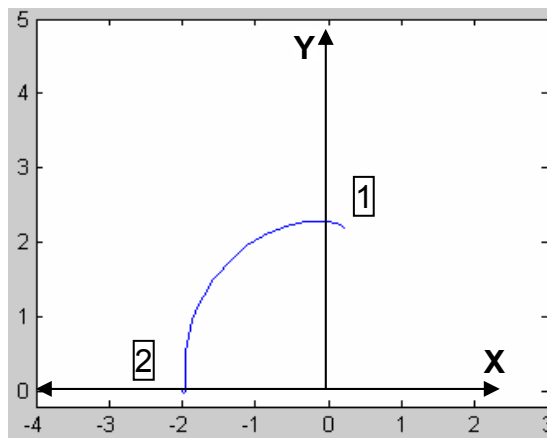
**Figura 3.28** Esquema de control PID.

Los eslabones tienen una masa de 0.5 kg cada uno en las extremidades. Se utilizó simulink® 4.0 para simular los torques que generan los actuadores, que en este caso serán servomotores de corriente alterna. Esta información obtenida nos permitirá seleccionar el par (torque) necesario de los motores para realizar la tarea sin problemas de desempeño. Las ganancias se seleccionaron de tal forma que el error tendiera a cero. La técnica que se utiliza fue la de par calculado con el conocimiento del modelo dinámico del robot.

En la tabla 3.3, se presentan las ganancias utilizadas.

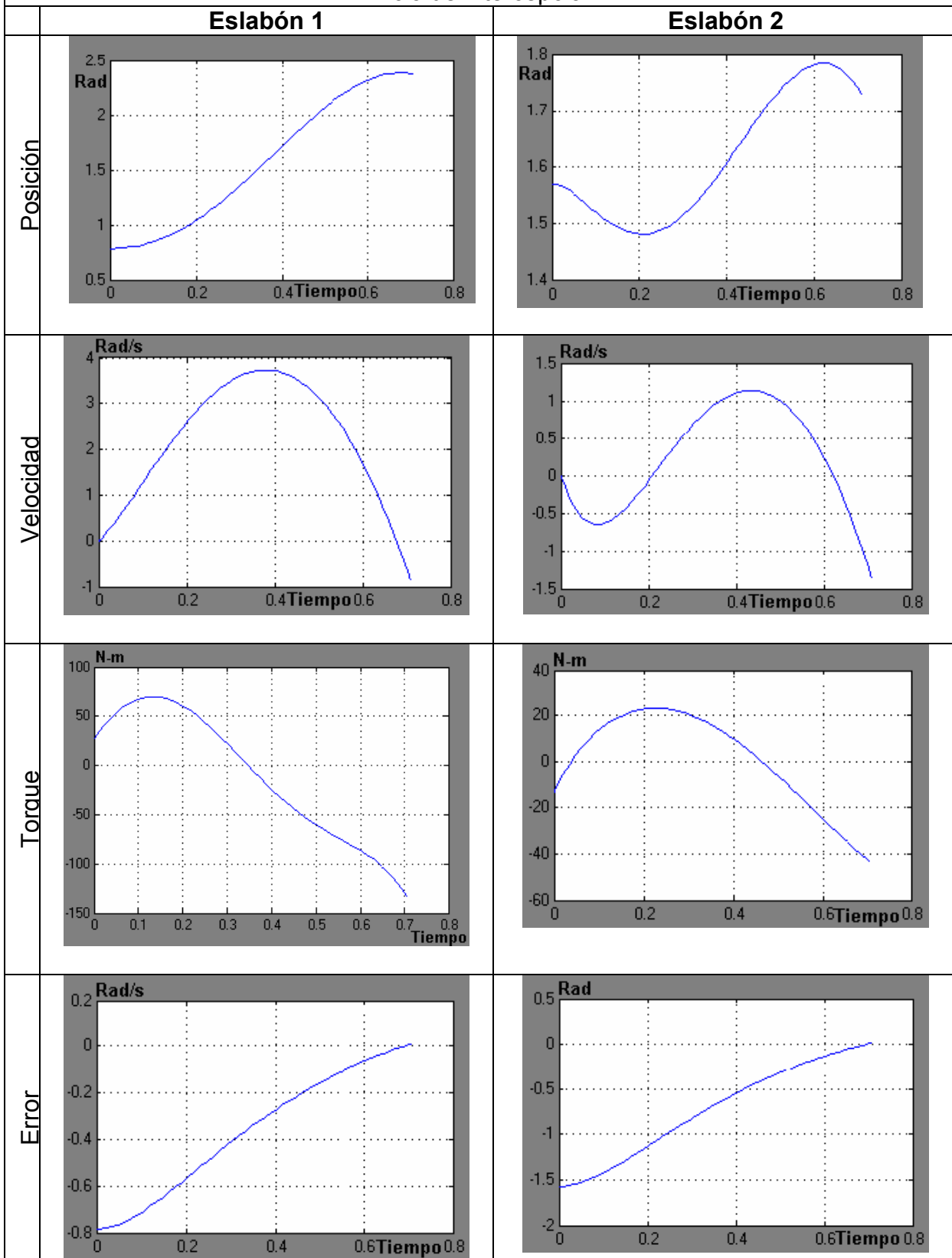
<b>Tabla 3.3</b> Ganancias $K_p$ , $K_v$ y $K_i$ seleccionadas			
Posición	Ganancias	Eslabón 1	Eslabón 2
Inicio intercepción	$K_p$	25	25
	$K_v$	10	10
	$K_i$	16	16
Posición 1	$K_p$	15	15
	$K_v$	5	5
	$K_i$	2.5	2.5
Posición 2	$K_p$	25	25
	$K_v$	10	10
	$K_i$	14	14

**Posición de inicio de intercepción:** En la figura 3.28, se presenta la trayectoria generada de la posición 1 a la 2 del efector final del robot interceptor, en 0.7 s. (Ini\_inicio.mdl)

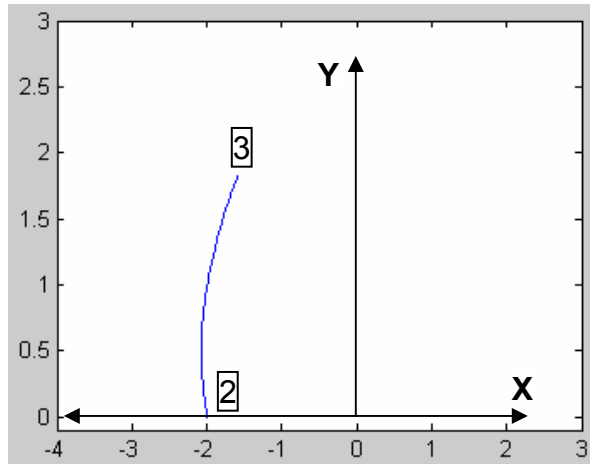


**Figura 3.29** Trayectoria generada por el efector final del robot interceptor a la posición de inicio de intercepción. (-2.023, 0.03753).

**Tabla 3.4** Gráficas posición y velocidad de los eslabones 1 y 2 del robot interceptor. Inicio de intercepción

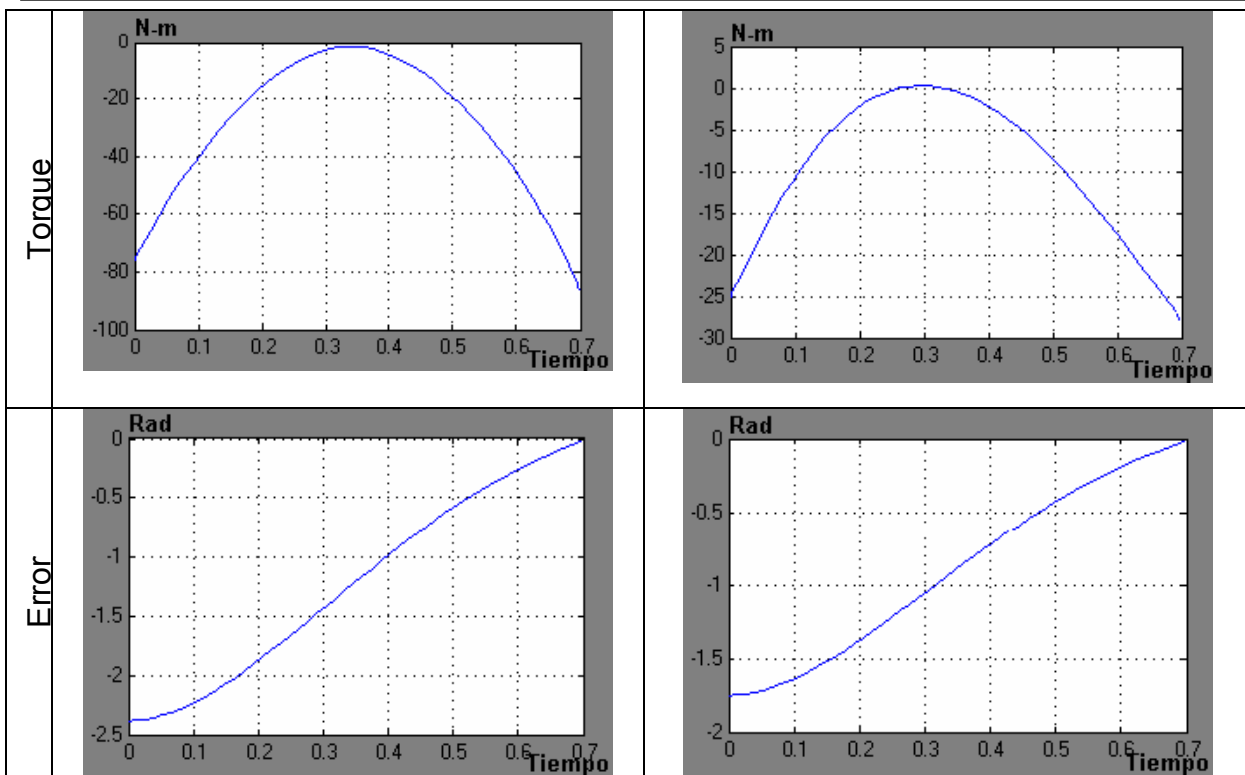


**Posición 1:** En la figura 3.29, se presenta la trayectoria generada de la posición 2 a la 3 del efector final del robot interceptor, en 0.7 s. Cuando intercepta la primer partícula TPL1.

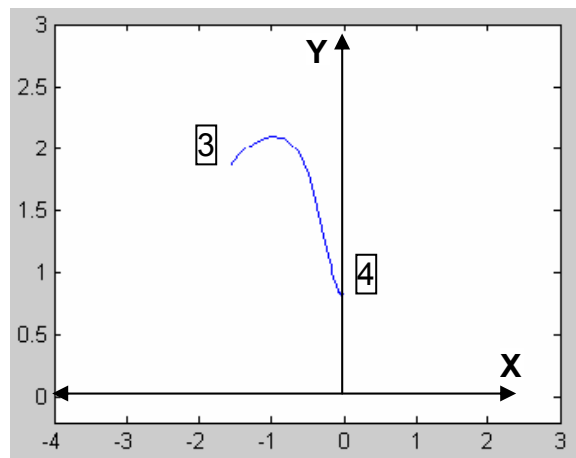


**Figura 3.30** Trayectoria generada por el efector final del robot interceptor a la posición 2, en este caso punto 3. (-1.576, 1.836)

		Tabla 3.5 Gráficas posición y velocidad de los eslabones 1 y 2 del robot interceptor. Posición 1.	
		Eslabón 1	Eslabón 2
Posición			
	Velocidad		

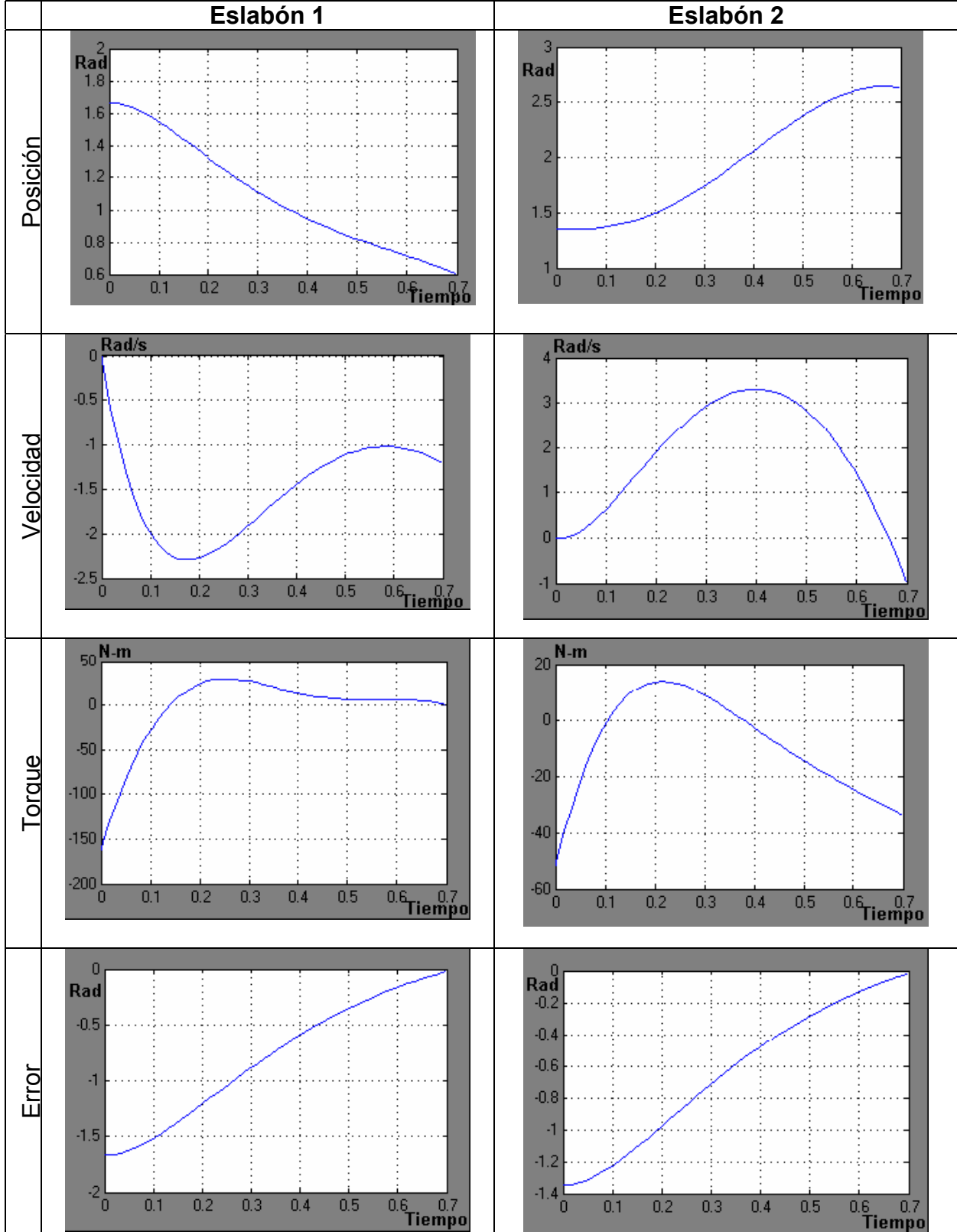


**Posición 2:** En la figura 3.31, se presenta la trayectoria generada de la posición 3 a la 4 del efector final del robot interceptor, en 0.7 s. Cuando intercepta la segunda partícula TPL2.



**Figura 3.31** Trayectoria generada por el efector final del robot interceptor a la posición 1, en este caso punto 3. (0.003622, 0.8472)

**Tabla 3.6** Gráficas posición y velocidad de los eslabones 1 y 2 del robot interceptor.  
Posición 2.





En la tabla 3.6, se presentan las gráficas de los torques para que los servos motores realicen la tarea satisfactoriamente. Se observa que el torque del eslabón 1 es el mayor, de la posición 3 a la 4 su valor es de -160 N-m aproximadamente, tres veces más que el torque para el eslabón 2, esto es lógico por que el eslabón 1 mueve a los dos eslabones en su movimiento individual, el eslabón dos se mueve sólo.

En la tabla 3.6, se muestran también las posiciones angulares en el tiempo de los servomotores que mueven los eslabones 1 y 2, aquí se puede observar que tienen un comportamiento parabólico lo que da como resultado un movimiento suave.

### III.17 SUMARIO

Se realizaron simulaciones con valores numéricos de los algoritmos de intercepción dando resultados satisfactorios. Se resolvió el problema de la intercepción de una partícula cuyo evento inició el origen de los sistemas coordenados que coincidían.

En el fenómeno de intercepción múltiple se logró acoplar la cinemática del robot interceptor de dos grados de libertad con la trayectoria que genera la partícula seguidora TPS para interceptar las trayectorias de las partículas libres TPL1 y TPL2, de tal forma que se adecuaron los instantes de integración para los dos subsistemas: el robot y las trayectorias de las partículas libres y la partícula seguidora.

El sistema de control PID que se implementó resultó satisfactorio parcialmente. Pero fue necesario implementar la parte integrativa I, del controlador para eliminar el error en estado permanente, siendo las ganancias propuestas las que mejores resultados dieron.

Los principales logros que se alcanzaron en este capítulo son tres; el primero es que se validaron los algoritmos de recuperación de tiempo en simulación comprobando que si funcionan. En los ejemplos de simulación se propusieron valores numéricos diferentes, con el fin de mostrar el comportamiento de la trayectoria TPS, también se variaron los tiempos de intercepción y los ángulos de disparo de las partículas libres.

El segundo logro fue el desarrollo de un simulador para la intercepción de dos partículas libres, este permitirá realizar eventos de tal forma que se pueda elegir la trayectoria adecuada a las condiciones y restricciones físicas.

El tercer logro, es el acoplamiento de la trayectoria TPS al la cinemática inversa del robot interceptor de dos grados de libertad, y al planteamiento del esquema de control a utilizar en la parte experimental.

Algunos detalles como son: la obtención de las coordenadas relativas de las articulaciones de robot y la obtención del modelo dinámico se consideraran y serán de utilidad en el siguiente capítulo.

## III.18 REFERENCIAS

- [3.1] Russell L. Anderson, "A Robot Ping Pong Player: Experiments in Real-Time Intelligent Control", Cambridge, MA: MIT Press, 1988.
- [3.2] Germán V. M., Cinemática y dinámica de la partícula, Instituto Politécnico Nacional, Dirección de Publicaciones, 2002. ISBN 970-18-8146-X.
- [3.3] F.L. Lewis, Control of Robot Manipulators, Micmillan Publishing Company, a division of Macmillan, Inc., 1993; ISBN 0-02-370501-9
- [3.4] A. Barrientos y otros, Fundamentos de Robótica, McGraw-Hill, 1997. ISBN 84-481-0815-9.
- [3.5] K.S. Fu y R.C. González. Robotics: Control, sensing, Vision and Intelligence, McGraw-Hill Book Company, 1987, ISBN 0-07-022625-3.
- [3.6] Kelly R., "Control de Movimiento de "Robots Manipuladores", Centro de investigación Científica y de Educación Superior de Ensenada México, 1994.
- [3.7] Spong M. y M. Vidyasagar, "Robot Dynamics and Control", John Wiley and sons, Primera Edición, New York, 1989.
- [3.8] Sciavicco L. y B. Siciliano, "Modeling and Control of Robot Manipulators", Mc Graw-Hill, Primera Edición, New York, 1996.
- [3.9] Lewis F.L., C.T. Abdallah y D.M. Dawson, "Control of Robot Manipulators", MacMillan Publishing Company, 1993.
- [3.10] Canudas de Wit C., B. Siciliano and Bastin (editores), "Theory of Robot Control", Springer-Verlag, Primera edición, London, 1996.
- [3.11] Kruse I., "Modeling and Control of a Flexible Beam and Robot Arm", Tesis de Doctorado, University of Twente, Enschede. The Netherlands, 1990.
- [3.12] Corke P.I. y Armstrong, "A meta-study of PUMA 560 dynamics: A critical appraisal of literature data", Robótica. Vol. 13. p: 253-258, 1995.
- [3.13] Canudas C., P. Noel, A. Aubin y B. Brogliato, "Adaptative Friction compensation in Robot Manipulators: low Velocities". The international Journal of Robotics Research. Vol. 10. No.3 June. P: 189-199, 1991.
- [3.14] Southward S.C., C.J. Radcliffe y C.R. Maccluer, "Robust nonlinear Stick- Slip Friction Compensation". ASME Journal of Dynamics Systems, Measurement and Control. Vol. 113. p:639-645, 1991.
- [3.15] A. Flores. C. L. Héctor, H.G *et al*, "Cálculo Simbólico Formal de Manipuladores Robóticos". 3er. Congreso Internacional de Ingeniería Electromecánica y de Sistemas. IPN, p:99-105, 2002.
- [3.16] Craig J. J. "Introduction to Robotics: Mechanics and Control", Addison Wesley, Segunda Edición, Reading, Massachusetts, 1989.
- [3.17] Santibañez V., R. Kelly y F. Reyes, "A new set-point controller with IEEE Transactions on Industrial Electronics. 1997.
- [3.18] Takegaki M y S. Arimoto, "A new feedback method for dynamic control of manipurators", ASME J. Dyn. Syst. Meas. Control. Vol. 103, p: 119-125, 1981.
- [3.19] Paden B. y Panja, "Globally asymptotically stable PD+ controller for robot manipulators", International Journal of control. Vol. 47. No. 6. P: 1697-1712, 1988.
- [3.20] Slotine J.J. y W. Li, "Adaptative manipulator control: a case study", IEEE Transactions on Automatic Control, 33(11) p: 995-1003, 1988.

- 
- [3.21] An C.H., C.G. Atkeson and J.M. Hollerbach, "Modal-Based Control of a Robot Manipulator", The MIT Press, 1988.
- [3.22] An C.H., G. Atkeson, J.D. Griffiths y J.M. Hollerbach, "Experimental Evaluation of feedforward and computed torque control", IEEE Trans. On Robotics and Automation. Vol. 5. No. 3. June, p:368-373, 1989.
- [3.23] Khosla P. y T. Kanade, "Parameter identification of robot dynamics", En Proc. 24<sup>th</sup> IEEE Conf. ON Decision and Control. Fort Lauderdale FL., p: 1754-1760, 1985.
- [3.24] Khosla P. y T. Kanade, "Real-Time implementation and evaluation of computed-torque scheme", IEEE Trans. On Robotics and Automation. Vol. 5. No. 2. p: 245-252, April 1989.
- [3.25] Leahy M. B, "Modelling-Based control of industrial manipulators: An experimental analysis", Journal of Robotics Systems, Vol. 7. No. 5, p: 741--758, October 1990.
- [3.26] Lu Z., K.B. Shimoga y A.A. Goldenberg, "Experimental determination of dynamic parameters of robotic arms", Journal of Robotics Systems, 10(8), John Wiley & Sons. P: 1009-1029, 1993.
- [3.27] Kelly R., "Control de Movimiento de Robots Manipuladores", Centro de Investigación Científica y de Educación Superior de Ensenada, México, 1994.
- [3.28] Anibal Ollero Baturone, "Robótica Manipuladores y Robots Móviles", Alfa Omega Marcombo, 2001. ISBN 84-267-1313-0.
- [3.29] Richard M. Murray, "A Mathematical Introduction to Robotic Manipulation", CRC Press, 1994. International Standard Book Number 0-8493-7981-4
- [3.30] Mehran Mehrahdz, Naftali M. Sela, Robert G. Fenton y Beno Benhabib, "Robotic Interception of Moving Objects Using an Augmented Ideal Proportional Navigation Guidance Technique", IEEE Transactions on Systems, Man and Cybernetics- part a: Systems and Humans, vol. 30, No. 3, p:238-250, Mayo 2000.
- [3.31] Germán V. M., "Cinemática y dinámica de la partícula", Instituto Politécnico Nacional, Dirección de Publicaciones, 2002.

---

## CAPÍTULO IV

### ANÁLISIS EXPERIMENTAL

#### IV.1 GENERALIDADES

Antes de aplicar los algoritmos de recuperación de tiempos al Sistema de Manufactura Flexible (FMS) y a un prototipo de robot manipulador de 2 gdl, para validar la función seguidora TPS experimentalmente, es necesario disponer de los dispositivos que intervendrán para realizar los eventos experimentales, éstos son los que se muestran en la figura 4.1.

El inciso a), se desarrolló en el capítulo II, en el inciso b) se desarrolla la función seguidora TPS que intercepta dos objetos sin impactarlos, es decir, en algún instante de tiempo  $\Delta t$  la velocidad de la función TPS es igual a la velocidad de la función de las trayectorias de las partículas TPL1 y TPL2, respectivamente.

La metodología para desarrollar la función seguidora considera como información dos tipos de movimiento c) Lineal y d) tipo parabólico. Para el caso del robot interceptor, inciso j), éste se diseña y se construye. Luego, se plantea un esquema de control PID, inciso f) para controlar su movimiento articular y lograr que el efector final del robot siga la trayectoria TPS en términos de posición y velocidad.

Para medir el ángulo de disparo y la velocidad de caída de las esferas que el robot debe interceptar, se diseña un dispositivo de disparo múltiple con ángulos ajustables y un sensor de velocidad con el fin de controlar el movimiento de las partículas, inciso h).

Para el caso del Sistema de Manufactura Flexible (FMS), se desarrolla el modelo cinemático del robot Mitsubishi®, además, se desarrolla el sistema de visión para medir la posición relativa y la velocidad de las piezas que se mueven sobre la banda transportadora, inciso i).

La obtención del modelo cinemático se desarrollará para los dos robots, el prototipo y el robot industrial, inciso e). Ambos casos experimentales se realizarán para un evento particular. Esto significa que solo se tomarán los datos y condiciones iniciales para realizar un evento experimental, y que la implementación para cualquier evento queda para trabajos futuros.

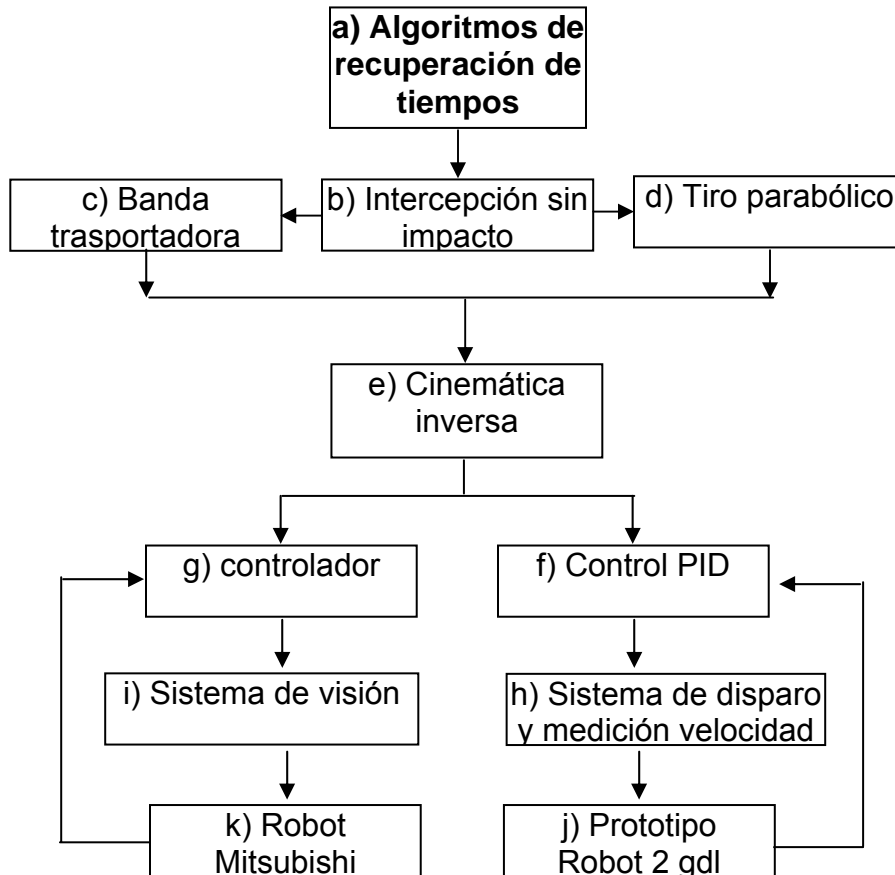


Figura 4.1 Esquema del planteamiento del trabajo experimental

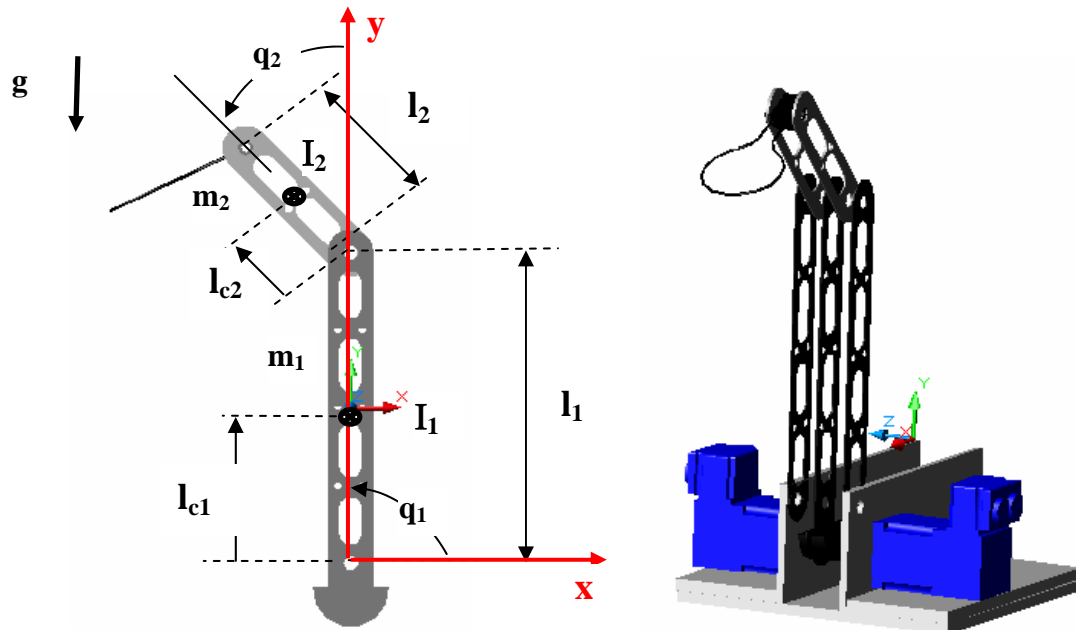
## IV.2 ROBOT MANIPULADOR (ROBOT INTERCEPTOR)

De acuerdo a la definición adoptada por el Robot Institute of America (RIA): Un robot es un manipulador multifuncional, reprogramable, diseñado para mover materiales, partes, herramientas o dispositivos especializados a través de movimientos programados para la ejecución de una variedad de tareas. Esta definición data de 1980, reflejando el estado actual de la tecnología en robótica [4.1].

Una vez definidas las dimensiones del prototipo de robot interceptor de dos grados de libertad, en la parte de simulación computacional. Ahora, en este capítulo se presentará la parte experimental. Además, se realizarán pruebas previas del control de movimiento del robot interceptor con el fin de prepararlo.

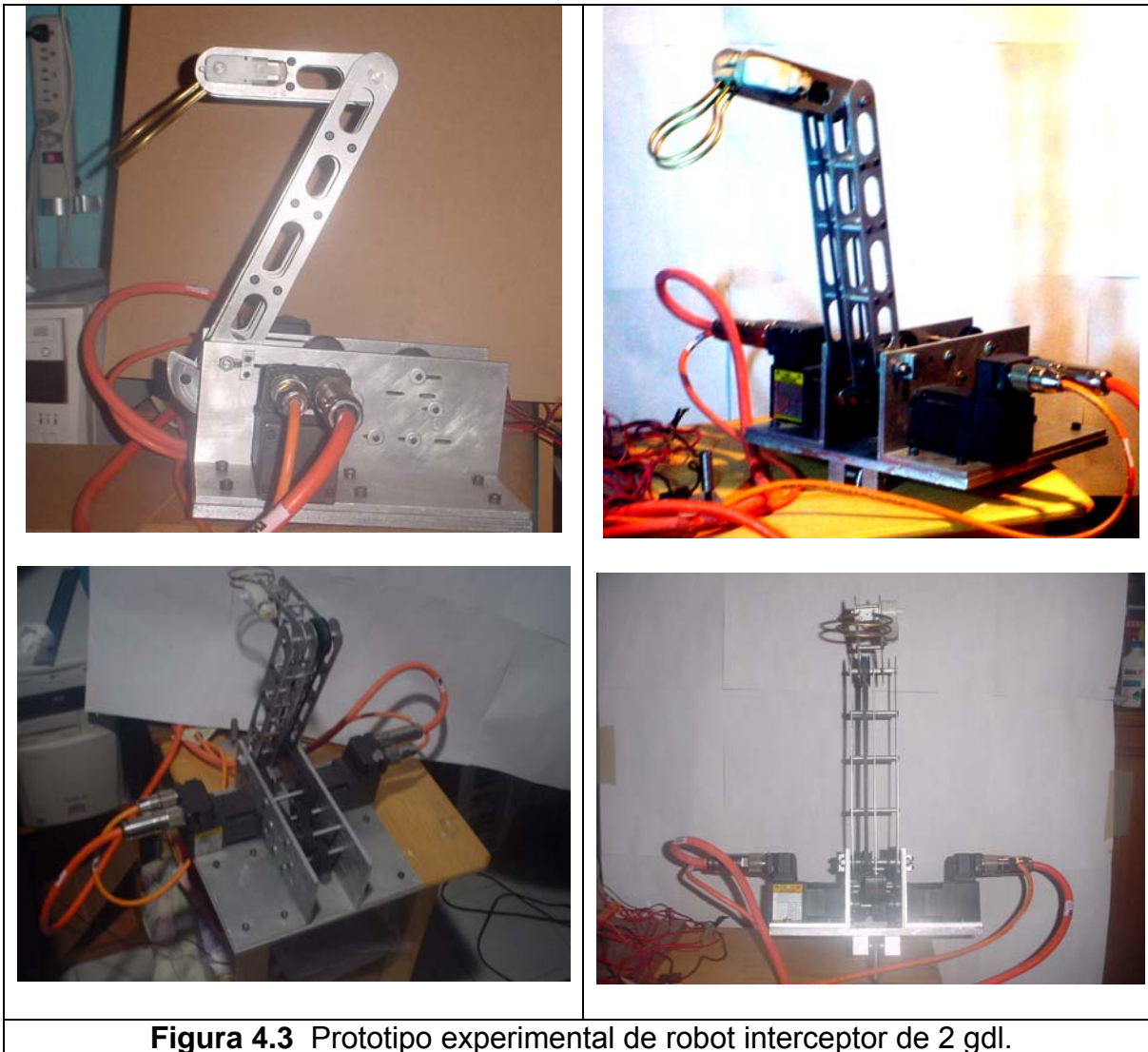
A continuación se describe brevemente algunos aspectos del robot manipulador propuesto. Este robot experimental es de transmisión por banda dentada para el codo y para el hombro. Está construido de placas delgadas de aluminio perforadas para disminuir el peso en los eslabones y obtener mayor rigidez, los motores se encuentran en la base, ver figura 4.2.

El robot interceptor de efector final tiene una argolla destinada a capturar los objetos en pleno vuelo. Lo anterior significa, que no se considera la forma del gripper ya que no esta dentro de los alcances de este proyecto.



**Figura 4.2** Esquema de prototipo de robot interceptor de 2 gdl.

El propósito de ofrecer el estudio práctico del fenómeno de intercepción, es comprobar experimentalmente los algoritmos desarrollados en el capítulo II por un lado, y por el otro lado, presentar la diferencia que existe entre la simulación y la parte real del fenómeno. Además, de que en la parte experimental existen factores que no son considerados en la simulación, como son: la fricción, o el modelo matemático exacto del robot manipulador(real), entre otros [4.2,4.4]. Al realizar la parte experimental se valida la parte teórica, la cual no incluye aspectos como son el ruido en los sensores, la dinámica de los actuadores, las fricciones, etc. Con lo anterior, se asegura el éxito en el mundo real de las aplicaciones, siendo los resultados de simulación limitados [4.5].



**Figura 4.3** Prototipo experimental de robot interceptor de 2 gdl.

Un robot de transmisión directa se define como un brazo mecánico donde todas sus articulaciones son manejadas con transmisión directa. Consiste en que el rotor del motor esté directamente acoplado al eslabón, eliminándose el sistema de transmisión por engranaje. Dando como resultado una reducción en el fenómeno de fricción, y eliminación de falta de movimiento (Backslach). La construcción es simple y la exactitud en el posicionamiento del robot manipulador es mejorada [4.6].

Por otro lado, los robots convencionales utilizan trenes de engranes para amplificar el par de sus actuadores con un factor  $\beta$ , multiplicando la inercia de los rotores un factor  $\beta^2$ , causando que la inercia del rotor domine sobre la inercia de los eslabones, permitiendo que la dinámica no lineal del robot pueda ser ignorada. Además, el sistema de engranaje produce fenómenos de elasticidad y de fricción. Los pares debidos a la fricción pueden ser considerables sobre la dinámica del robot. Por ejemplo, un robot PUMA 600 tiene un par debido a la fricción de hasta el 50% del máximo par del motor [4.6].

Considerando lo anterior, el prototipo de robot interceptor tiene una transmisión convencional. Es decir, cuenta con un sistema de engranaje para cada eslabón y los motores descansan sobre la base del robot (ver figura 4.3), esto es debido a que la velocidad nominal del servomotor es de 7000 rpm, que es bastante alta para desplazamientos angulares pequeños de los eslabones. Los desplazamientos angulares de los eslabones son menores a  $180^\circ$ . Esto obedece a que el robot solo tiene un área definida de trabajo en el eje positivo "X". A continuación se presentan las características mecánicas y eléctricas del robot interceptor de dos grados de libertad moviéndose en el plano vertical, con articulaciones rotacionales y los eslabones se consideran rígidos, ver tabla 4.1.

<b>Tabla 4.1</b> Notación de los parámetros físicos de la Figura 4.3.		
<b>Eslabón</b>	<b>Significado</b>	<b>Notación</b>
<b>1</b> (HOMBRO)	Masa del eslabón 1	$m_1$
	Longitud del eslabón 1	$l_1$
	Inercia del eslabón 1	$I_1$
	Centro de masa del eslabón 1	$l_{c1}$
	Posición articular del eslabón 1	$q_1$
<b>2</b> (CODO)	Masa del eslabón 2	$m_2$
	Longitud del eslabón 2	$l_2$
	Inercia del eslabón 2	$I_2$
	Centro de masa del eslabón 2	$l_{c2}$
	Posición articular del eslabón 2	$q_2$
	Aceleración debida a la gravedad	$g$

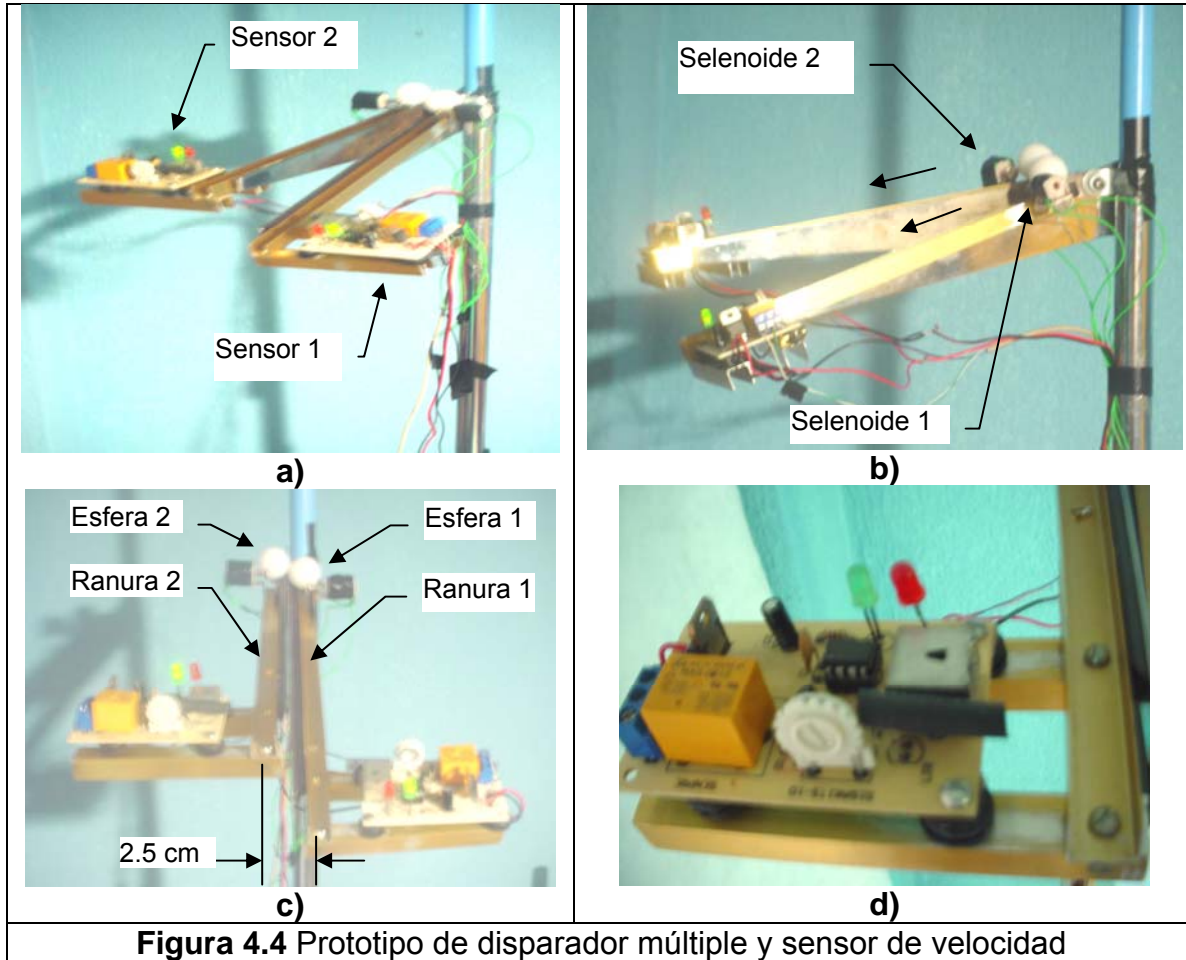
Cabe mencionar, que el robot interceptor tiene una argolla en su extremo que es el efector final, y permite capturar los objetos con una red. El área de la argolla es de  $40 \text{ cm}^2$ , aproximadamente. Los objetos esféricos, aunque se encuentran separados en dos planos paralelos una distancia de  $0.025 \text{ m}$  ( $2.5 \text{ cm}$ ) logran ser capturados por la argolla.

### IV.3 DISPARADOR DE OBJETOS MÚLTIPLE

En la figura 4.4, se muestra el dispositivo propuesto para realizar los eventos de intercepción de dos objetos vía robot interceptor, fue necesario construir un disparador que es capaz de lanzar dos objetos esféricos desde una altura de  $2 \text{ m}$  aprox. medido desde el piso. Los objetos esféricos son liberados por un selenoide cada uno por separado y descansan sobre la ranura, ver figura 4.4b. El selenoide es activado a través de que emite una señal de  $+24 \text{ volts}$ . Además, cuenta con un sensor al final de la ranura que mide el tiempo que tarda en pasar la esfera, esto con el fin de calcular la velocidad de salida de los objetos. La esfera tiene un diámetro de  $0.025 \text{ m}$  ( $2.5 \text{ cm}$ ), y su peso es de  $102 \text{ gramos}$ .

Las ranuras se pueden ajustar independientemente de tal forma que el ángulo de disparo de los objetos es variable. La señal de selenoide llega al mismo tiempo, ya que se utiliza una sola señal para activar los selenoides [4.8].





**Figura 4.4** Prototipo de disparador múltiple y sensor de velocidad

#### IV.4 PARÁMETROS DINÁMICOS DEL ROBOT INTERCEPTOR 2 GDL.

Los parámetros dinámicos del robot están compuestos por masas, distancias desde el centro de masa al eje de giro y momentos de inercia con respecto al centro de masa de cada eslabón, los cuales dependen de la composición física de cada eslabón. El eslabón 1 está compuesto por el rotor del motor BSM 50N 133AA de la marca Baldor®, así como los tornillos que sujetan a los motores. Los parámetros del primer eslabón se calculan a partir de las siguientes ecuaciones:

$$m_1 = m_{b1} + m_{r1} + m_{estator2} + m_{tornillos1} + m_{tornillos2} \quad (4.1)$$

$$I_{c1} = \frac{m_{b1}l_{cmb1} + m_{estator2}l_1 + m_{tornillos2}l_1}{m_1} \quad (4.2)$$

$$I_1 = I_{b1} + I_{r1} + I_{estator2} + m_{estator2}l_1^2 \quad (4.3)$$

donde  $m_{b1}$  es la masa de la barra 1,  $m_{r1}$  es la masa del rotor 1,  $l_{cmb1}$  es la distancia del centro de masa al eje de giro de la barra 1,  $m_{estator2}$  es la masa del estator 2,  $m_{tornillos1}$  es

la masa de los seis tornillos que acoplan las tres barras del hombro. Los valores numéricos de los parámetros dinámicos del robot interceptor se muestran en la tabla 4.2 y tabla 4.3.

<b>Significado</b>	<b>Notación</b>	<b>Valor</b>	<b>Unidades</b>
Longitud 1	$L_1$	0.252	m
Masa barra 1	$m_{b1}$	0.205	kg
Masa rotor 1	$m_{r1}$	0.045	kg
Masa estator 2	$m_{estator2}$	0.400	kg
Masa tornillos1	$m_{tornillos1}$	0.0067	kg
Masa tornillos2	$m_{tornillos2}$	0.0067	kg
Centro de masa barra 1	$L_{cmb1}$	0.126	m
Inercia barra 1	$L_{b1}$	0.000271	kg m <sup>2</sup> /rad
Inercia rotor 1	$L_{r1}$	0.000078	kg m <sup>2</sup> /rad
Inercia estator 2	$L_{e1}$	0.000095	kg m <sup>2</sup> /rad

<b>Significado</b>	<b>Notación</b>	<b>Valor</b>	<b>Unidades</b>
Longitud 2	$L_2$	0.200	m
Masa barra 2	$m_{b2}$	0.145	kg
Masa rotor 2	$m_{r2}$	0.0045	kg
Masa tornillos3	$m_{tornillos3}$	0.0067	kg
Centro de masa Barra 2	$L_{cmb2}$	0.100	m
Inercia barra 2	$L_{b2}$	0.000121	kg m <sup>2</sup> /rad
Inercia rotor 2	$L_{r2}$	0.000078	kg m <sup>2</sup> /rad

#### IV.5 REQUERIMIENTOS DEL ORDENADOR

Los componentes necesarios para completar la instalación del servosistema se presentan abajo, debe ser un ordenador con las siguientes especificaciones, ver tabla 4.4.

<b>Recurso</b>	<b>Especificaciones mínimas</b>	<b>Especificación recomendada</b>
Procesador	Intel Pentium 133 MHz	Intel Pentium 200 MHz o más.
RAM	32 MB	64 MB
Disco duro	40 MB	60 MB
CD-ROM	DRIVE de CDROM	
Pantalla	800 x 600, 256 colores	1024 x 768, 256 colores
Ratón	Un ratón o un dispositivo similar	
Sistema operativo	Windows 95, Windows 98, Windows ME, Windows NT, Windows 2000 o Windows XP	
Ranura de PCI	Debe haber una ranura disponible en la Mother Board de la PC tipo PCI.	

El equipo que se utilizó para desarrollar el prototipo de robot interceptor, fue una computadora personal Pentium II, con sistema operativo Windows 98, con procesador Intel Pentium 450 MHz y una resolución de la pantalla de 800 x 600, 256 colores.

## IV.6 DESCRIPCIÓN GENERAL DE LOS SERVOMOTORES DEL ROBOT INTERCEPTOR.

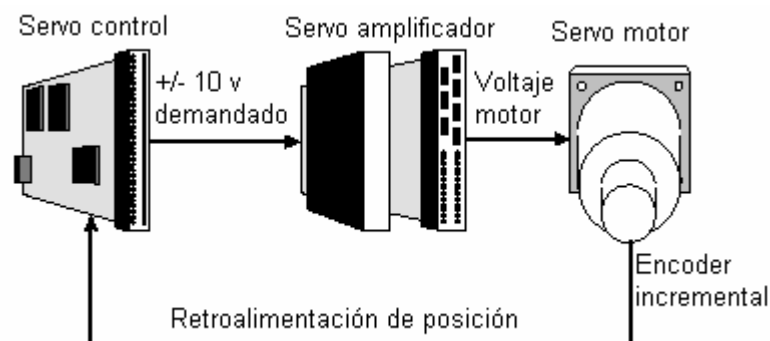
A continuación se presenta una breve descripción del equipo que se utilizó para el control de los servomotores. El servoamplificador es de la marca Baldor Flex+Drive® para usarse con servomotores sin escobillas. Este servocontrol digital utiliza entradas y salidas de  $\pm 10\text{VCC}$ , La marca del motor es del mismo fabricante y es de la serie BSM-A. El sistema se comunica a la PC a través del protocolo de comunicación RS-232 y se utiliza una tarjeta de adquisición de datos para establecer comunicación entre los amplificadores y la computadora. Se utilizan encoders incrementales de cuadratura ópticos de 1024 líneas, obteniendo una resolución de 4096 pulsos por revolución para retroalimentar la posición de los motores al servocontrol.

### IV.6.1 SISTEMA DE SERVO POSICIONAMIENTO

Un sistema que se utilizó para realizar la parte experimental fue de posicionamiento de lazo cerrado y éste se puede dividir en tres partes:

- 1.- **Servo control.**- Se encarga del control de posición en tiempo real del servomotor, guarda el programa de aplicación, se comunica con el usuario y con otros equipos de control.
- 2.- **Servo amplificador.**- Toma las señales de demanda del servo control de posición para controlar el torque o la velocidad del servo motor.
- 3.- **Servo motor.**- Transforma la corriente eléctrica que proviene del servo amplificador en movimiento rotatorio o lineal. El motor tiene un sensor de posición (Encoder incremental), que alimenta la salida para cerrar el lazo de retroalimentación.

El servo control trabaja muestreando la posición del motor en intervalos regulares, comparando esta posición con la posición deseada. Ordena al amplificador que mueva el motor para corregir el error en posición. Este proceso se repite generalmente 500 veces por segundo para asegurar que el motor se encuentre en la posición correcta siempre. En la figura 4.5, se muestra un sistema típico de control de posición en lazo cerrado.



**Figura 4.5** Sistema típico de control de posición en lazo cerrado.

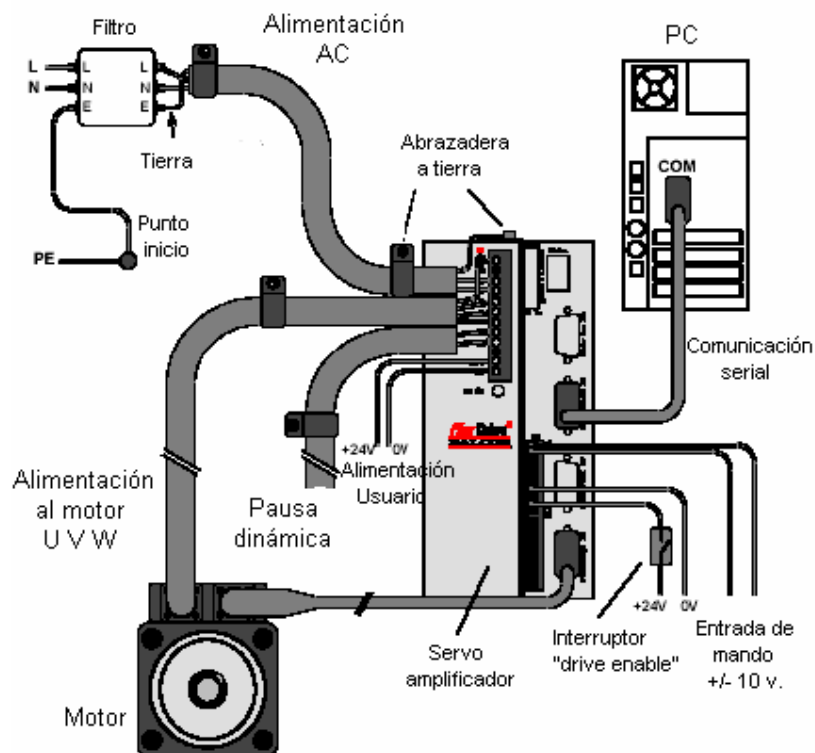
## IV.7 PONER EN MARCHA EL SERVOSISTEMA.

El servo control puede funcionar con muchos servo sistemas, sean hidráulicos o eléctricos, mientras trabajen con una entrada de control de  $\pm 10$  v, como demanda de torque o velocidad, y que generen una salida de retroalimentación de la posición en forma de encoder incremental. Un sistema mínimo de alambrado es uno, en el cual el alambrado del servo controlador y el servo amplificador es configurado con el menos alambrado posible. En la figura 4.6, se muestra el sistema mínimo de alambrado de un eje (un servo motor y un servo amplificador) que se utilizó en este trabajo.

El servosistema FlexDrive<sup>II</sup> mostrado en la figura 4.6, puede configurarse básicamente en tres modos de control de operación. El modo deseado se selecciona desde el programa.

- Control por corriente (Torque)
- Control de velocidad (Velocidad)
- Control de posición (Pulso y dirección )

A continuación se presenta una breve descripción de los modos de control del sistema utilizado.



**Figura 4.6** Configuración mínima del servo sistema.

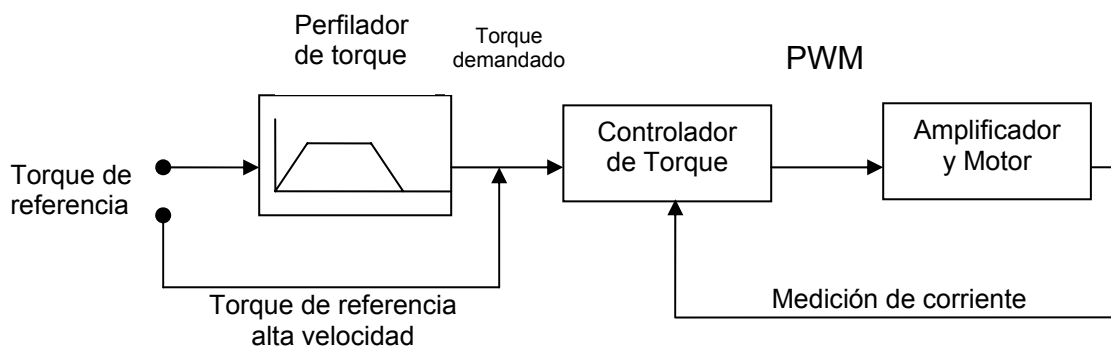
### IV.7.1 CONTROL POR CORRIENTE (TORQUE)

Fijando este modo de control se configura el servocontrol como un amplificador de torque (par de torsión), como se muestra en la figura 4.7. Aquí la referencia del torque se obtiene de una fuente específica. Estas son:

- Señales de comando desde el ordenador personal (con perfil).
- Entradas analógica de alta velocidad (sin perfil).
- Entrada analógica (con Perfil)
- Fieldbus (con perfil)

La fuente suministra una señal que puede ser alimentada directamente al controlador de torque (sólo cuando se usa una fuente de entrada analógica de alta velocidad), o a través del perfilador. La fuente de referencia a alta velocidad es ideal para conectarse a un controlador de movimiento que tendrá ya perfilada la señal dando como resultado un control de lazo cerrado óptimo.

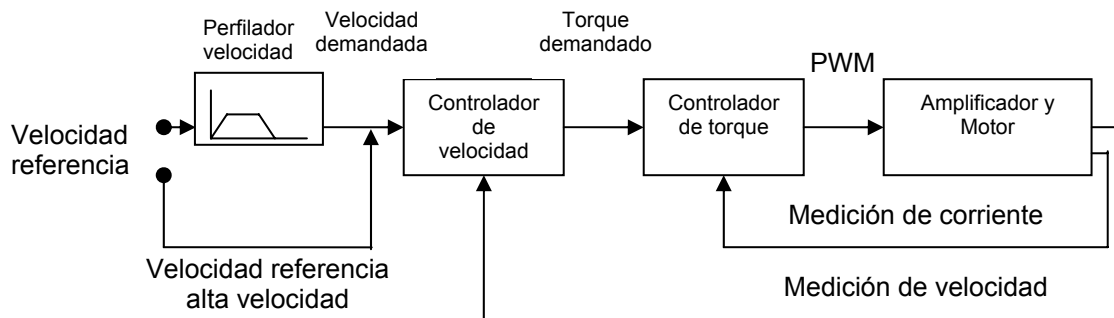
El perfilador (generador de señales o perfiles de movimiento) genera una señal de torque demandado que cambia suavemente entre los sucesivos objetivos de torque (valores de referencia). Esto se logra especificando un tiempo de elevación y un tiempo de caída en el programa de aplicación. La señal de torque demandado es enviada al controlador de torque, el cual determina la cantidad de corriente que se debe aplicar a los devanados del motor. Esta corriente demandada es comparada con la actual corriente demandada medida desde los sensores, y un pulso de PWM (modulación de ancho de pulso) apropiado se genera para enviarlo a la electrónica de potencia del amplificador.



**Figura 4.7** Estructura de control en el modo de control por corriente (torque).

### IV.7.2 CONTROL DE VELOCIDAD

Fijando el modo de control de velocidad configura el FlexDrive<sup>II</sup> como amplificador de velocidad, como se muestra en la figura 4.8. Aquí, la velocidad de referencia se obtiene de una fuente específica, que son los mismos que para el control por corriente.



**Figura 4.8** Estructura de control en el modo de control de velocidad.

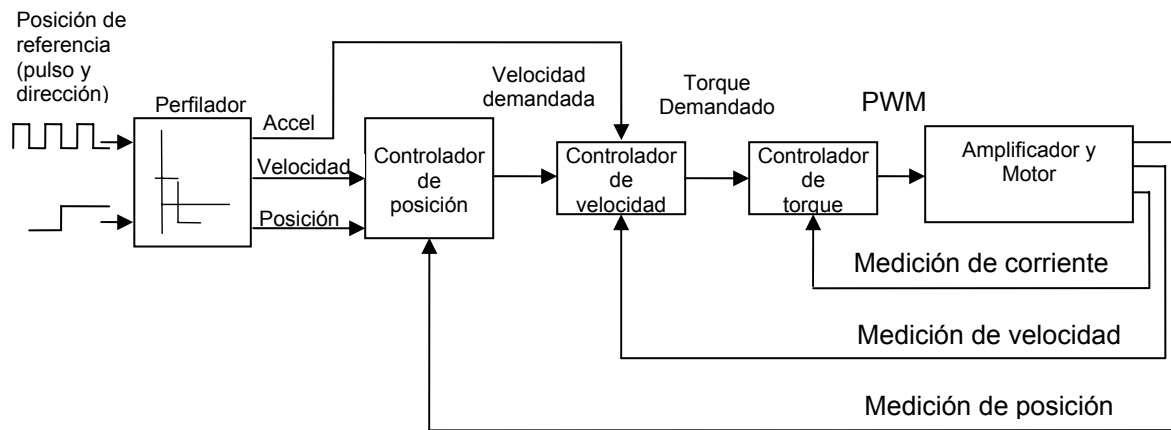
La fuente suministra una señal que puede ser alimentada directamente al controlador de velocidad (sólo cuando se usa una fuente de entrada analógica de alta velocidad), o a través del perfilador. La fuente de referencia de alta velocidad es ideal para conectarse a un controlador de movimiento, que tendrá ya perfilada la señal dando como resultado un control de lazo cerrado óptimo.

El perfilador genera una señal de velocidad demandada con cambios suaves entre los sucesivos objetivos del perfil de velocidad (valores de referencia). Esto se logra especificando un tiempo de aceleración y un tiempo de desaceleración en el programa de aplicación. La señal de velocidad demandada es enviada al controlador de velocidad y usada, junto con la velocidad medida del dispositivo de retroalimentación, para generar una señal de torque demandado. Si el controlador de velocidad es sintonizado correctamente, la velocidad leída seguirá la trayectoria de la velocidad demandada con exactitud.

Finalmente, la señal de torque demandado es enviada al controlador de torque, este determina la cantidad de corriente que se debe aplicar a los devanados del motor. La corriente demandada es comparada con la corriente demandada por los devanados del motor, medida con los sensores, y un pulso de PWM (modulación de ancho de pulso) apropiado se genera para enviarlo a la electrónica de potencia del amplificador.

### IV.7.3 CONTROL DE POSICIÓN

Fijando el modo de control a control de posición (pulso y dirección), configura al FlexDrive<sup>II</sup> como un sistema de posicionamiento, como se muestra en la figura 4.9, capaz de seguir un comando de posiciones.



**Figura 4.9** Estructura de control de posición (pulso y dirección).

El perfilador interpreta las señales de pulso y dirección y los utiliza para generar la posición, velocidad y aceleración demandadas.

Las señales de posición y velocidad demandadas son enviadas al controlador de posición y usadas, juntas con la posición medida del dispositivo de retroalimentación, para generar una señal apropiada de velocidad demandada. Si el controlador de posición está sintonizado correctamente, la posición medida seguirá la trayectoria de la posición demandada con exactitud.

La señal de velocidad demandada proveniente del controlador de posición es enviada al controlador de velocidad y usadas, con la velocidad medida del dispositivo de retroalimentación, para generar la señal de torque demandado.

Si el controlador de velocidad está sintonizado correctamente, la velocidad medida seguirá la trayectoria de velocidad demandada con exactitud. Para mejorar el desempeño de seguimiento de la velocidad del controlador, el perfilador de aceleración demandada se añade en este punto.

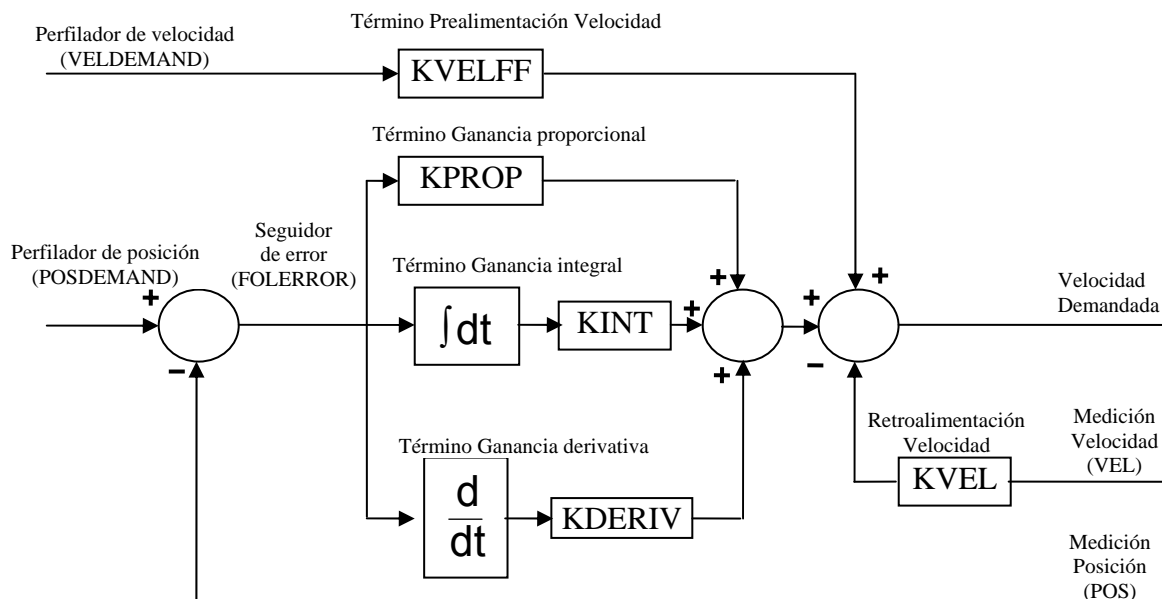
Finalmente, la señal de torque demandado entra al controlador de torque el cual determina la cantidad apropiada de corriente a aplicar a los devanados del motor. Esta corriente demandada es comparada con la corriente de los devanados actual del motor medida por los sensores. Esta señal de PWM es enviada a la electrónica del amplificador.

#### IV.7.4 OPERACIÓN DEL SISTEMA DE CONTROL

A continuación se describe la operación de los controladores de posición, velocidad y torque.

## IV.7.4.1 CONTROLADOR DE POSICIÓN

El controlador de posición, mostrado en la figura 4.10, es un típico controlador proporcional+integral+derivativo (PID), en el cual las ganancias son dadas por las palabras clave de programación: KPROP, KINT y KDERIV. La posición demandada del perfilador es comparada con la posición leída y el error generado es enviado al cálculo del control PID. Los valores forman la velocidad base o el torque demandado, dependiendo del ajuste de la configuración. El efecto del término integrativo puede cambiarse usando las palabras clave KINTLIMIT y KINTMODE. KINTLIMIT fija el valor máximo del efecto de la acción integral, como un porcentaje de toda la escala demandada. KINTMODE puede usarse para especificar las circunstancias bajo las cuales el término integral es aplicado.



**Figura 4.10** Sistema de control de posición.

Como una alternativa para el término derivativo, un término de retroalimentación de velocidad está disponible con la palabra clave KVEL. Para mejorar el desempeño del seguimiento de la trayectoria de posición, el controlador de velocidad es prealimentado por la ganancia KVELFF, para dar una contribución a la velocidad final o al torque demandado. El perfilador de aceleración es prealimentado por la ganancia KACCEL para dar una contribución a la velocidad final demandada. El cálculo del control de posición se ejecuta cada  $500\mu s$ . Los valores de las ganancias KPROP, KINT, KDERIV, KVEL, KVELFF y KACCEL deben estar sintonizadas para cada aplicación. Se pueden fijar las ganancias por programación utilizando el lenguaje de programación del paquete WorkBench<sup>®</sup> v5, que es el programa que contiene el controlador y el Firmware<sup>®</sup> del sistema de movimiento.



#### IV.7.4.2 CONTROLADOR DE VELOCIDAD

El controlador de velocidad mostrado en la figura 4.11, que fue el que se utilizó en la parte experimental, es también un controlador del tipo PID. Las ganancias se pueden programar usando las palabras clave KVPROP, KVINT y KVDERIV. La velocidad es comparada con la velocidad medida y el error se envía al cálculo del control de PID. Los resultados forman el torque demandado para el controlador de torque. Para mejorar el desempeño del término derivativo, se incluye en forma opcional un filtro. La constante de tiempo del filtro se puede fijar usando las palabras clave KVDERIVTCONST. Por *default* el filtro está desactivado, con KVDERIVTCONST igual a cero. Cuando el sistema de movimiento FlexDrive<sup>II</sup> es activado en modo de posicionamiento (CONFIG es \_cfSERVO), el perfilador de aceleración es prealimentado con la ganancia KACCEL para dar una contribución al torque final demandado. El cálculo del control de velocidad se ejecuta cada 250µs. Así como para el controlador de posición, los valores de las ganancias KVPROP, KVPROP, KVDERIV y KVACCEL deben ser sintonizados para cada aplicación. Estas ganancias pueden ser dadas por el Commissioning Wizard<sup>®</sup> o en forma manual usando la herramienta de Fine-Tuning<sup>®</sup> en el programa de Work-Bench<sup>®</sup> v5.

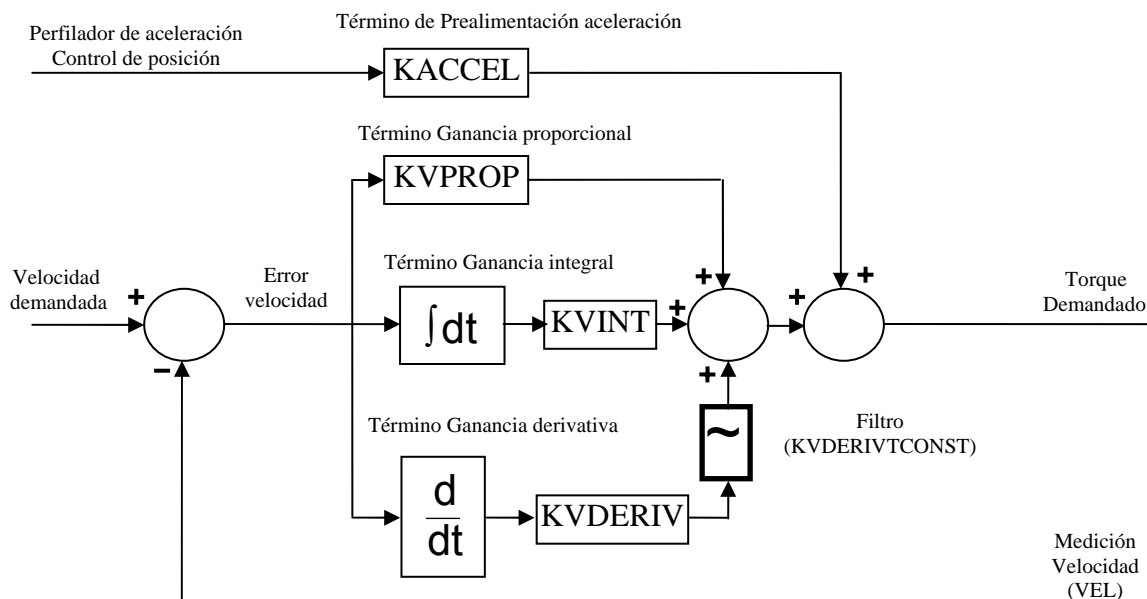


Figura 4.11 Sistema de control de velocidad.

#### IV.7.4.3 EL CONTROLADOR DE TORQUE Y LA RETROALIMENTACIÓN

El controlador de torque mostrado en la figura 4.12 es un controlador PI. Las ganancias se fijan usando las palabras clave KIPROP y KIINT. El torque demandado es adaptado en la corriente demandada. Esta es comparada con la corriente medida obtenida de los sensores de corriente y el error es introducido al cálculo del control PI. El valor que se obtiene es transformado en una señal de PWM que es enviado a la electrónica de

potencia del amplificador dentro de los devanados del motor. Esto se logra automáticamente utilizando el Commissioning Wizard®.

Aunque el control PI puede alcanzar el seguimiento de corriente adecuadamente, si éste está sintonizado correctamente, el desempeño se incrementa usando retroalimentación en modelo de voltaje. Este modelo compensa la caída de voltaje debido a la resistencia del estator (se fija usando la palabra clave MOTORRS) y la inductancia que se fuga del estator (palabra clave MOTORLS), y también compensa la fuerza electromotriz EMF del motor, con la constante del voltaje del motor se usa la palabra clave MOTORFLUX. Los valores para MOTORRS, MOTORLS y MOTORFLUX se fijan automáticamente por el Commissioning Wizard®. El cálculo del control del torque se ejecuta cada 125µs.

El dispositivo de retroalimentación (por ejemplo, encoder o resolver) se utiliza para determinar la posición del motor y velocidad. La velocidad del motor se puede filtrar para reducir en la medición el ruido si es necesario. La constante de tiempo de este filtro se especifica usando la palabra clave KVTIME. Por *default* el filtro está apagado (KVTIME=0). Es importante mencionar que introducir el filtro en la medición de la velocidad tiende a reducir la estabilidad del controlador de velocidad. Esto puede hacer muy difícil la sintonización de las ganancias si los valores de KVTIME son grandes.

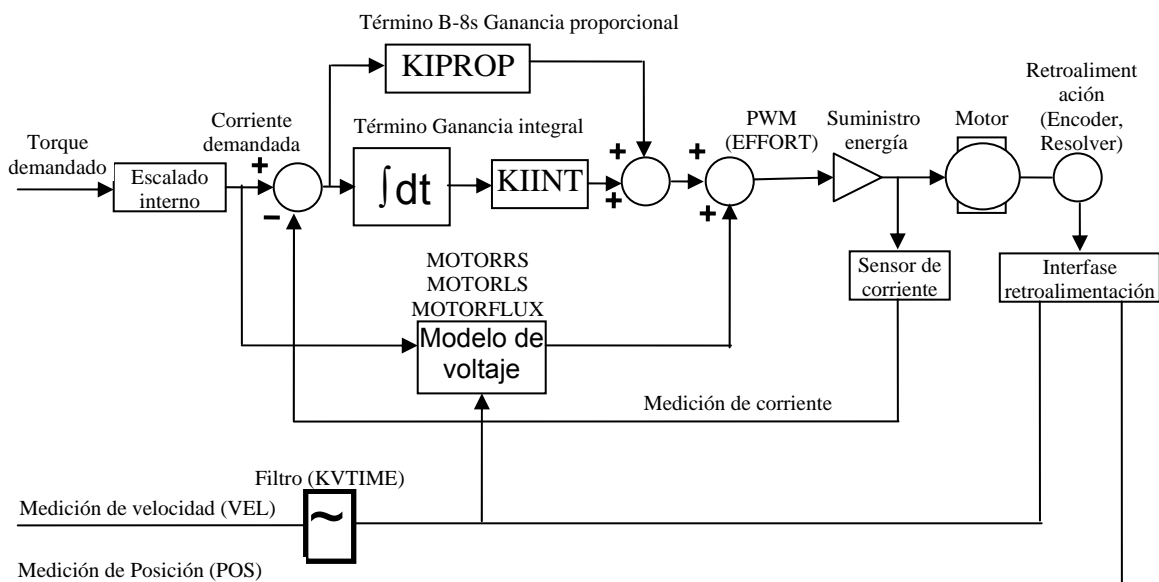


Figura 4.12 Sistema de control por torque.

## IV.8 EL SERVO CONTROLADOR

El controlador NextMove® se diferencia fundamentalmente de otros controladores de movimiento debido a la alta velocidad con que cuenta el procesador digital de señales de punto flotante. Desde que los procesadores pueden procesar eficientemente números de punto flotante, muchos de los Firmware® (microprograma lógico

inalterable), incluyendo los algoritmos del lazo de control usan números de punto flotante. Esta es la mayor ventaja de la generación de los perfiladores de movimiento. Ya que todos cálculos pueden ser llevados a cabo en unidades significativas antes de la conversión a unidades de máquina (cuentas de encoder de cuadratura, pasos) dentro del algoritmo de control de lazo cerrado. La parte de tiempo real del código consiste de dos manejadores de rutinas de interrupción:

- Cada 2 ms (500 Hz), el algoritmo perfilador genera datos de puntos de referencia para todos los ejes(servomotores) de movimiento.
- Cada 1000  $\mu$ s (1 kHz) el controlador muestrea las posiciones, calcula demanda instantánea de los ejes sobre las bases de la interpolación lineal de la salida del perfilador y genera una demanda para el servo corriendo el lazo PIDFA, a lo anterior se le llama Lazo de control.

Además, existe una rutina de interrupción de 2 ms, la cual desempeña las tareas de mantenimiento internas, actualizando el puerto dual RAM y el control de motores a pasos. En el nivel más bajo del programa de control, las demandas de posición del eje instantáneas causadas por el programa del controlador deben ser convertidas en las demandas de motor. Esto es conseguido por el bucle de control cerrado del motor. El motor es controlado de tal forma que minimiza el error entre la demanda y la posición actual medido con un codificador incremental.

Cada vez que el lazo de control es actualizado, el controlador compara las posiciones actuales con las deseadas y calcula la demanda correcta para el motor. El torque es calculado por el algoritmo PIDFVA (Proporcional, Integral, Derivativo, Retroalimentación de Velocidad, prealimentación de velocidad y aceleración).

El control puede ser alcanzado aplicando sólo torque proporcional al error, pero éste es un enfoque algo simplista. Imagina que existe un error pequeño entre la posición actual y la demandada, un controlador proporcional simplemente multiplicará el error por alguna constante, la ganancia proporcional, y aplicando el resultado a un motor a través de un amplificador.

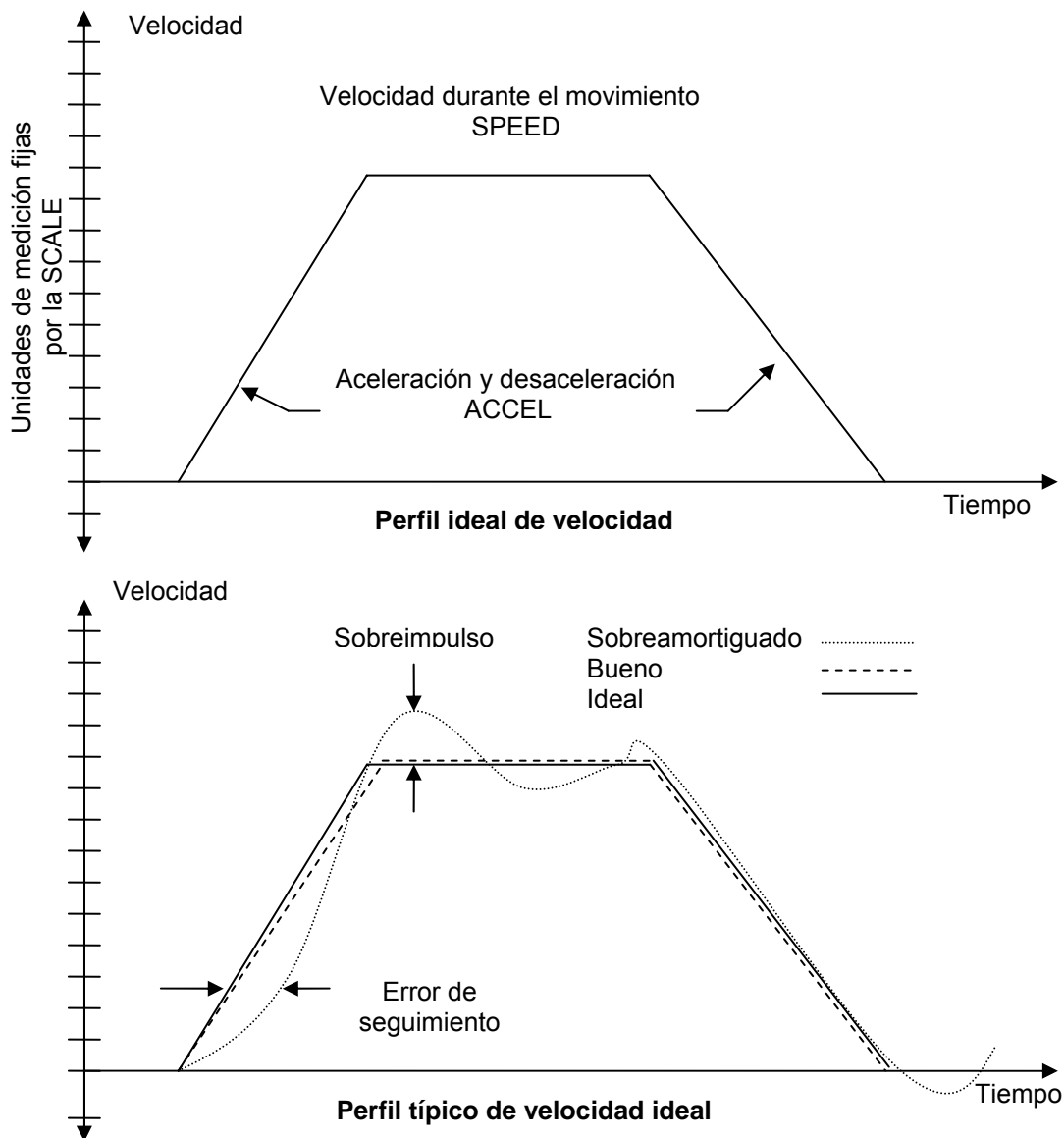
Si la ganancia es demasiado alta causará un sobre impulso, el cual causará que el motor vibre de un lado a otro de la referencia o posición deseada. Al incrementar la ganancia, el controlador presentará mayor resistencia al error de posición, pero las oscilaciones crecerán en magnitud hasta volver el sistema inestable.

Para reducir el inicio de inestabilidad, se incorpora un término de amortiguamiento al algoritmo de lazo de control, llamado ganancia de retroalimentación de velocidad. La retroalimentación de velocidad actúa para resistir movimientos rápidos del motor. Por lo tanto, permite fijar una ganancia proporcional alta antes de que la vibración empiece. (En algunas aplicaciones la retroalimentación de velocidad es manejada por el servo amplificador, llamada servo velocidad).

El efecto de una alta ganancia proporcional, o una baja ganancia de retroalimentación de velocidad es ilustrada con la línea sobreamortiguada en la figura 4.13.

Un método alternativo de amortiguamiento es suministrado en la forma de derivada de la señal del error. La acción derivativa tiene el mismo efecto que la retroalimentación de velocidad, si la retroalimentación de velocidad es igual a la prealimentación de velocidad. En los sistemas de control de torque la acción derivativa es el término preferido. Con las acciones proporcional y derivativa es posible que un motor tenga un error de posición pequeño llamado "following error" (error de seguimiento).

El controlador multiplica el error con término de ganancia proporcional para producir un torque correctivo (en control de corriente), pero para errores muy pequeños el torque puede no ser tan grande como para vencer la fricción estática. Este error puede ser corregido incorporando un término integral en los cálculos del lazo. La acción integral involucra la suma del error en el tiempo, de tal forma que el torque del motor crece gradualmente hasta que el error de posición se vuelve cero.



**Figura 4.13** Perfil de velocidad trapezoidal ideal y típico.

La velocidad a la cual la acción integrativa trabaja es controlada por la ganancia intergral. La acción integral es útil para eliminar los errores en estado permanente. Pero reduce la respuesta dinámica del sistema.

Por esta razón, se provee la opción seleccionable por medio del programa donde el usuario puede seleccionar cuando apagar el integrador durante el movimiento. El último término en el lazo de control es la prealimentación de velocidad. Este es útil para incrementar la respuesta y reducir el error de seguimiento, especialmente en los servos de control de velocidad. Existen dos tipos de servo amplificadores que pueden ser usados con el controlador:

- Amplificador de torque o corriente utilizan la señal de demanda para controlar el flujo de corriente en la armadura del motor y por lo tanto el torque del motor.
- El amplificador de control de velocidad (servo velocidad) utiliza la señal demandada como referencia de servo velocidad.

Para aplicaciones generales, el amplificador de torque es económico y simple prepararlo, pero el servo de velocidad desempeña mejor control especialmente en aplicaciones de alto desempeño.

Para los amplificadores de torque la prealimentación de velocidad debe ser usada para estabilizar el sistema, pero no es normalmente requerido para el servo de velocidad desde que se ha incorporado internamente su propia retroalimentación de velocidad.

La figura 4.14, muestra el diagrama del lazo de control completo, incluyendo el controlador, el amplificador, el motor y el reductor de velocidad (engranaje). El servo amplificador puede ser un amplificador de corriente simple, o incorporar la retroalimentación de velocidad internamente a través de un tacómetro.

El controlador tiene incorporados cinco términos, las ganancias; Proporcional, Derivativo, Integral, Retroalimentación y prealimentación de velocidad.

La ecuación del algoritmo de control de lazo cerrado es la siguiente:

$$\text{Demanda} = K_p \cdot e + K_d \cdot \frac{\Delta e}{\Delta \tau} - K_v \cdot v + K_F \cdot V + K_I \cdot \Sigma e + K_A \cdot a \quad (4.4)$$

donde;  $K_p$  (palabra de programación: KPROP) es la ganancia proporcional del lazo de control,  $K_D$  (KDERIV) es la derivada del error ganancia),  $K_v$ (KVVEL) es la ganancia de retroalimentación de velocidad,  $K_F$ (KVELFF) es la ganancia de prealimentación de velocidad,  $K_A$ (KACCEL) es la ganancia de prealimentación de aceleración,  $K_I$ (Integral de retroalimentación),  $e$  es el error,  $v$  es la velocidad,  $\tau$  es el tiempo de muestreo o el periodo de actualización del servo,  $v$  es la demanda de velocidad que está dado en cuentas de cuadratura/ tiempo de muestreo,  $a$  es la demanda de aceleración, (KINTMODE) es el modo de integración durante el movimiento, (KINTLIMIT) es el rango límite para el porcentaje del integrador máximo a la salida, (DACLIMITMAX)

Restringe el valor máximo de salida del DAC (Digital Analog Converter), (CURRENTLIMIT) restringe la máxima corriente de salida.

Para sintonizar el lazo de control es necesario dar valores a las ganancias  $K_p$ ,  $K_D$ ,  $K_I$ ,  $K_V$ ,  $K_F$  y  $K_A$  para suministrar el mejor desempeño para una combinación de motor/encoder con carga inercial. La respuesta dinámica del sistema permite seleccionar las ganancias adecuadas.

## IV. 9 PARÁMETROS DE LOS MOTORES

A continuación se presentan los resultados experimentales de la presintonización (Finetuning) de los motores, así como también, sus características principales de operación. La prueba fue hecha sin carga, es decir, el motor se probó sin tener acoplado a su eje ninguna carga, excepto la inercia del mismo eje.

Los dos motores que se utilizaron fueron del tipo: BSM 50N 133 AA, que manejan una corriente continua de 1.49 Amps (RMS), y una corriente pico límite 5.07 Amps (RMS) a una velocidad máxima 7000 rpm y Voltaje constante de operación de 30.38 volt/rpm. La Inductancia es de 16.5 Mh y la resistencia de sus bobinas es de 11.95 Ohms.

El drive que se utilizó tiene las siguientes características: es del tipo FDH1A02TB-RN20 y opera con un voltaje de 160 volts con corrientes pico de hasta 5 Amps (RMS) y una corriente nominal 2.5 Amps (RMS).

El Resolver que se utilizó tiene una resolución de  $1024 \times 4 = 4096$  cuentas por rev. Y un encoder  $1024 \times 4 = 4096$  cuentas por rev.

El modo de operación fue control de velocidad utilizando una entrada analógica para el control directo de velocidad.

Para la presintonización del motor se utilizaron los siguientes parámetros: Un tiempo de aceleración 100 ms, un tiempo de desaceleración 100 ms el error permitido en el tiempo de aceleración fue de 10 ms y la velocidad máxima permitida: 1200 rpm, se fijó la sobre-velocidad permitida a un 110% de la velocidad máxima 7000 rpm.

En la prueba de posición se manejó un factor de escala de 114 y un error de posición permitido de 0.399902 unidades de usuario (uu) con una tolerancia 0.244141 unidades de usuario. La velocidad entre puntos de movimiento: 1.2207 uu/uu y el ancho de banda calculado para el control de velocidad fue de 150 rad/s y el ancho de banda para el control de corriente fue: 2000 rad/s

Los resultados experimentales de la presintonización fueron los siguientes:

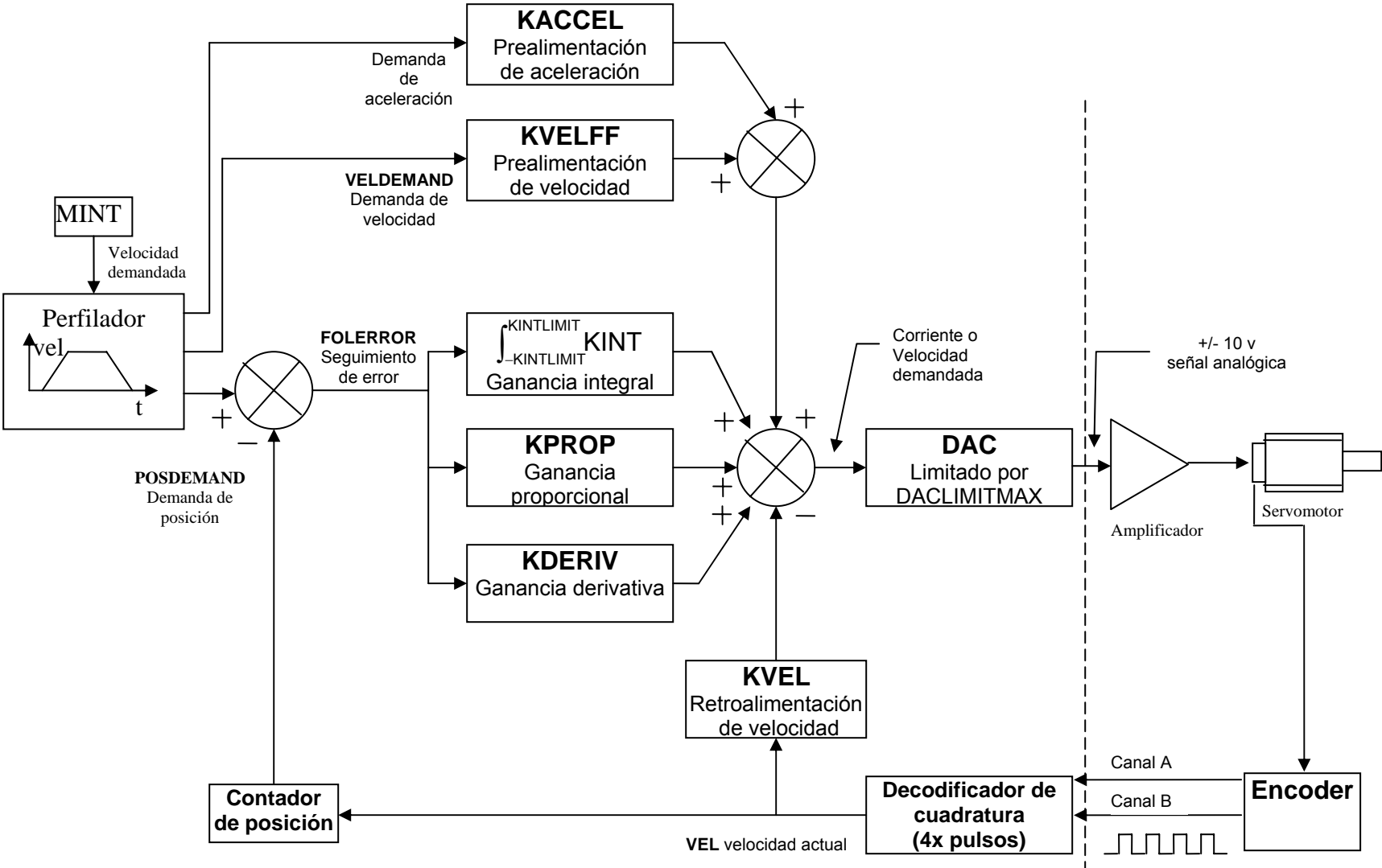


Figura 4.14 Esquema del Lazo de Control

Los valores reales medidos de los motores fueron: resistencia del motor: 13.647719 ohms y la Inductancia de 16.176313 mH. las ganancias calculadas por corriente fueron: KIProp: 0.761719 y KIInt: 1021.386719. En la prueba del offset de retroalimentación fue de -90.395508 degs y el voltaje constante: 28.113487 Vpk/kRPM.

El cálculo de las ganancias de posición y velocidad fueron las siguientes: KVProp: 0.597656, KVInt: 48.933929, KProp: 27.835938, KInt: 0.000000, KDeriv: 0.000000, KvelFF: 25.050781, Inercia del eje: 0.00001476 kgm<sup>2</sup>, el amortiguamiento: 0.00034273 Nms/rad y la velocidad: 50% a una distancia de 10 uu con una duración: 1.0 s.

A continuación se presentan las gráficas de la prueba de corriente y posición de los motores, ver figura 4.15.

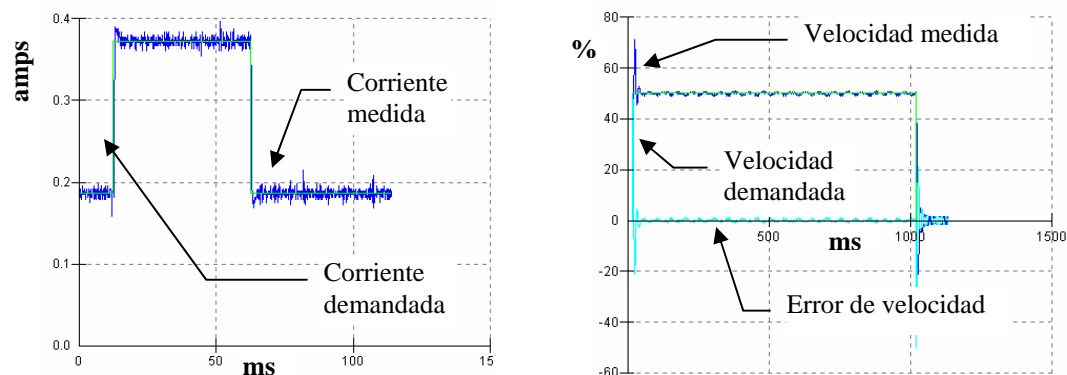


Figura 4.15 Resultados de control de corriente y velocidad , respectivamente.

## IV.10 ANÁLISIS DE RESULTADOS DEL SERVOCONTROL

Antes de realizar el experimento para que el robot intercepte las dos esferas fue necesario la presintonización, para evitar que los errores en el seguimiento de la trayectoria TPS sean grandes. Por esta razón, se realizaron las pruebas anteriores y los resultados fueron satisfactorios, ver figura 4.15. Aunque, cuando el motor se somete a la misma prueba con carga los resultados varían en forma considerable, por lo cual, es necesario volver a sintonizar el sistema pero ahora considerando la carga, que en este caso la carga la forman el movimiento acoplado de los eslabones, junto con la inercia del tren de engranes y el efecto del movimiento del robot interceptor.

### IV.10.1 IMPLEMENTACION DE CONTROLADOR P ANALÓGICO PARA EL GRIPPER.

El gripper del robot interceptor es una canasta cuadrada de 4.5 x 8 cm para capturar las esferas. Este gripper tiene un control de posición con un controlador tipo p (proporcional), y permite orientarla para capturar las esferas sin golpearlas.



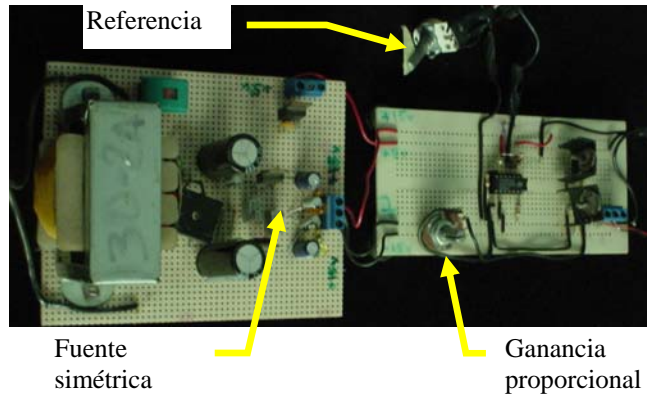
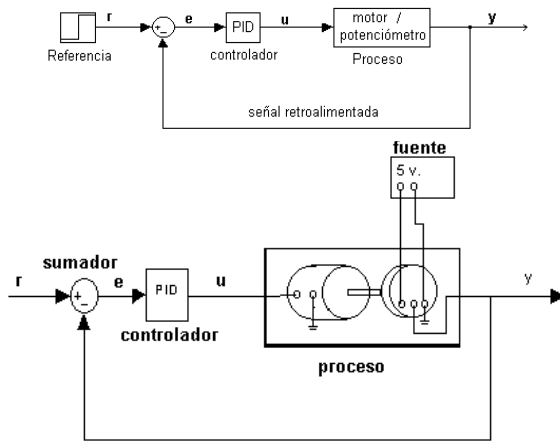


Figura 4.16. Diagrama de bloques del sistema controlado.

Figura 4.17 Foto de la electrónica del controlador p (proporcional)

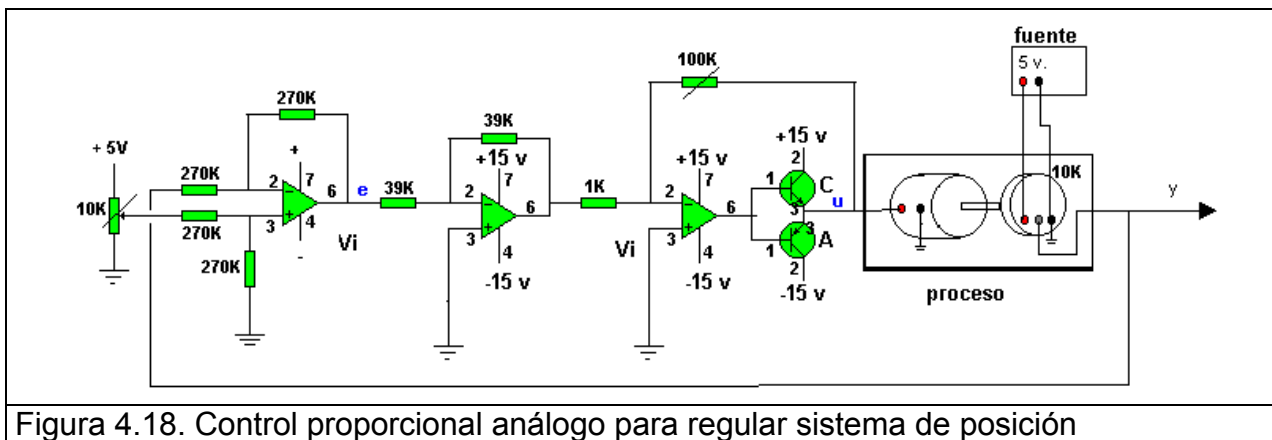


Figura 4.18. Control proporcional análogo para regular sistema de posición

$$E(s) \approx \frac{1}{1+K} R(s) \tag{4.5}$$

donde K es la ganancia del sistema en lazo abierto,

#### IV.10.2 ANÁLISIS DE RESULTADOS DEL CONTROLADOR PARA EL GRIPPER

Es evidente, que existirá error en la posición en estado permanente, sin embargo, para este caso de aplicación es suficiente este controlador. El controlador está conectado a una salida analógica de la tarjeta de adquisición de datos, desde ahí, se envía la señal de referencia en voltaje y ésta es traducida a posición angular del motor. Se realizaron algunos experimentos de voltaje y posición y los resultados fueron aceptables, aunque al principio de la operación del sistema la posición del gripper oscilaba entre la referencia enviada, se solucionó con la sintonización a través del potenciómetro hasta obtener una respuesta suave, sin golpeteo.

## IV.11 AMBIENTE DE PROGRAMACIÓN

En este tema se presentarán las palabras clave que se utilizan para dar valores a las ganancias a nivel de programación. A continuación, se describirán algunas de las palabras clave que se utilizarán en este proyecto:

**KPROP**, fija la ganancia proporcional para el controlador de posición, ejemplo de programación:  $KPROP = 1.5, 2.5$ . **KVDERIV**, este parámetro es usado a veces para reducir la oscilación y el sobre impulso en la respuesta del controlador de velocidad. Sin embargo, en muchas aplicaciones es suficiente usar los términos Proporcional e Integrativo (**KVPROP** y **KVINT**), el término derivativo tiene su propio filtro, la constante de tiempo se fija con la palabra clave **KVDERIVTCONST**. **KVINT**, igualando  $KVINT = 0$ , se remueve la compensación.

Esta selección es ideal para sistemas donde el sobreimpulso debe ser evitado y la rigidez (es la habilidad del controlador para mantener la velocidad a pesar de la variación de la carga) no es requerida. Incrementando esta ganancia incrementa la rigidez del control y mejora el desempeño del seguimiento de la velocidad.

Una excesiva ganancia del término integrativo causará un sobre impulso en el estado transitorio de la velocidad demandada pudiendo causar oscilación. **KVDERIVTCONST**, la constante de tiempo es usada como filtro en la parte derivativa del controlador de velocidad, es expresada en milisegundos. **KVELFF**, fija la ganancia de prealimentación de velocidad para el controlador de posición, ejemplo de programación:  $KVELFF = 10.5$ .

Para calcular el valor de **KVELFF**, es necesaria la siguiente información: La velocidad, en revoluciones por minuto producida por el motor cuando se aplica el máximo voltaje de (+10V) al amplificador. Para fijar el valor de **LOOPTIME** el fabricante lo definió a 1 ms (milisegundo). **LOOPTIME**, permite definir el intervalo de tiempo para actualizar el bucle del servo (lazo de control) en microsegundos.

El algoritmo de lazo cerrado es aplicado a todos los ejes de servo sobre una base periódica, y calcula el voltaje de salida requerido para los DAC's (Convertidor Analógico-Digital) con base en la posición actual medida y la posición requerida (demandada). La regularidad y la frecuencia de muestreo de este algoritmo afecta el desempeño de los servo amplificadores.

En términos generales, un alto desempeño de los servo amplificadores requieren de una frecuencia alta de actualización (bajos valores para **LOOPTIME**). El tiempo de respuesta del sistema físico es medido a menudo en milésimas de segundo, un ejemplo para fijar el bucle de servo a una milésima de segundo,  $1 \text{ ms} = \text{LOOPTIME} = 1000$

El sistema de movimiento Baldor® BSM utiliza encoders de 1024 ó 2500 líneas. La fórmula del bucle del servo utiliza valores de velocidad expresados en cuentas de cuadratura por bucle de servo. Para calcularlo: Primero, se divide la velocidad del motor en revoluciones por minuto, entre 60 para dar el número de revoluciones por segundo. Por ejemplo, si la velocidad del motor es 3000 rpm a la máxima velocidad demandada, entonces:  $\text{las rpm} = 3000 / 60 = 50$ . Luego, calcular cuantas revoluciones ocurren

durante un bucle de servo. El fabricante fija el tiempo del bucle de servo a un 1 ms (0.001 s), revoluciones por bucle de servo =  $50 \times 0.001 \text{ seg} = 0.05$

Ahora, se necesita calcular cuantas cuentas de encoder de cuadratura hay por revolución. Las cuentas de controlador vienen del encoder, de tal forma, que cada línea de encoder existen 4 cuentas de cuadratura con un encoder de 1000 líneas.

Cuentas de cuadratura por revolución =  $1000 \times 4 = 4000$

Finalmente, calcular cuantas cuentas de cuadratura hay por bucle de servo: Cuentas de cuadratura por bucle de servo =  $4000 \times 0.05 = 200$

La salida analógica demandada es controlada por un Convertidor Digital analógico (DAC), el cual crea voltajes de salida entre el rango de -10V a +10V. El valor máximo de voltaje de salida de +10V corresponde a una salida del Convertidor Digital Analógico (DAC) de 2048. El valor de KVELFF se calcula dividiendo 2048 por el número de cuentas de cuadratura por bucle de servo, dando:  $KVELFF = 2048 / 200 = 10.24$

**KVEL**, es la ganancia de prealimentación digital para velocidad de bucle de servo, ejemplo:  $KPROP = 0.1$ ,  $KPROP = 0.1$ ,  $KVEL = 2.3$ ,  $KVELFF = 2.7$ . **KACCEL**, para fijar la ganancia de prealimentación a el Bucle de servo de aceleración. Este término se utiliza para reducir los sobre saltos de velocidad en movimientos con altas aceleraciones, un ejemplo:  $KACCEL.0 = 45$  fija la ganancia de prealimentación a la aceleración a el eje 0. **KVPROP**, se utiliza para fijar la ganancia proporcional utilizada por el controlador de velocidad.

Este parámetro afecta la respuesta del controlador de velocidad demandada. Incrementando esta ganancia repercutirá en un respuesta rápida; un valor excesivo causará un sobre impulso y resonancia. El decremento de esta ganancia dará como resultado una baja respuesta y un decremento en el sobre impulso y en la resonancia.

**KIPROP**, es para fijar la ganancia proporcional utilizada en el controlador de corriente. Este parámetro se fija por Software en forma automática. **KIINT**, para fijar la ganancia integrativa utilizada en el controlador de corriente. Este parámetro se fija por programa en forma automática.

## IV.12 CÁLCULO DEL MOVIMIENTO TRAPEZOIDAL

Para que el movimiento del motor sea controlado es necesario entender los principios detrás del movimiento. El movimiento de objetos se puede especificar con varios parámetros, que juntos definen el perfil de movimiento (Motion Profile). Es usual definir el movimiento en términos de las siguientes propiedades: Distancia, Velocidad, aceleración, desaceleración y tiempo de movimiento.

Ejemplo: Se puede usar algunas ecuaciones de movimiento con aceleración uniforme, como:

---

$$v = u + at, \quad s = ut + \frac{1}{2} a t^2, \quad v^2 = u^2 + 2as \quad (4.6)$$

donde;  $u$  = velocidad inicial,  $v$  = velocidad final,  $a$  = aceleración y  $t$  = tiempo. Pero también se puede representar en forma gráfica:

Un ejemplo simple es un motor que tiene un encoder que genera 4000 cuentas por revolución y el eje es representado en grados, de tal forma que una revolución del motor genera 4000 cuentas de encoder y mueve la salida  $360^\circ$ . Por lo tanto, el factor de conversión será  $4000/360 = 11.11$  cuentas de encoder por grado. Así, la distancia, la velocidad y la aceleración serán en grados, grados por segundo, grados por  $s^2$  respectivamente.

Es importante considerar las unidades de medición durante todo el cálculo. Un encoder rotatorio usualmente se especifica por pulsos por segundo, debido al efecto conocido de cuadratura un controlador de movimiento lee 4 veces el número de cuenta. Esto es debido a que existen dos canales de salida (A y B), y cada pulso en cada canal tiene una bajada y una subida. Un encoder lineal usualmente se especifica en términos de distancia entre cada cuenta. Por ejemplo, 1  $\mu\text{m}$  (1 micrón), este valor normalmente no se multiplica por 4. Un 1  $\mu\text{m}$  tiene 1000 cuentas por mm de tal forma que el factor de escala debe fijarse a 1000 para cada mm de distancia.

#### IV.12.1 MOVIMIENTO DE LOS SERVOMOTORES

La distancia recorrida durante el movimiento debe ser especificada en Unidades de Usuario(UU), Esta distancia puede definirse en una de las dos formas, relativa o absoluta. Un movimiento relativo es definido como la distancia a recorrer desde la posición actual, y el movimiento absoluto es definido como un movimiento a una posición específica relativa a una posición predefinida como cero.

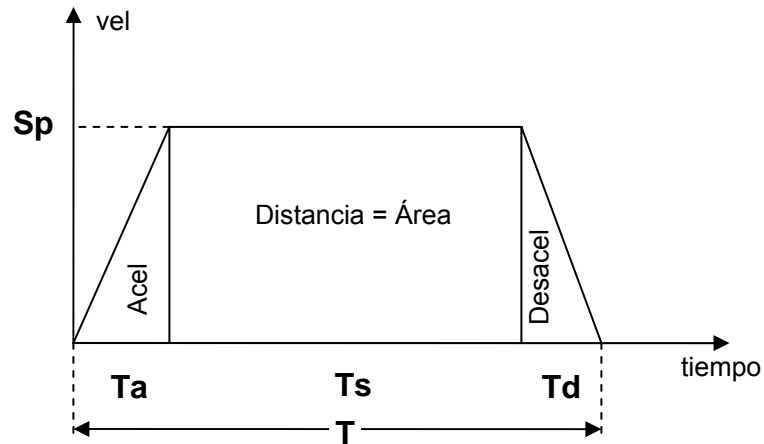
La distancia recorrida de un movimiento absoluto: es la posición final – posición inicial. Es importante entender que se refiere a la distancia recorrida, y no a donde se quiere mover.

Las palabras clave de programación MOVER y MOVEA son usadas para definir la posición del punto final, como movimiento relativo o absoluto, respectivamente. La palabra clave SPEED; se refiere a la velocidad conseguida después del tiempo de aceleración y está definida en Unidades de Usuario por segundo.

La aceleración es la razón de cambio entre unidad de tiempo de la velocidad y está definida en  $\text{UU}/s^2$ . Para el movimiento simple se asume una razón constante de aceleración, de tal forma que la razón de aceleración puede calcularse dividiendo la velocidad total entre el tiempo de aceleración. Las palabras clave de programación ACCEL y DECEL son usadas para fijar los valores de las razones de aceleración y desaceleración.

## IV.12.2 DIAGRAMAS DE VELOCIDAD-TIEMPO

Para hacer más fácil el entendimiento de los perfiladores de movimiento (generadores de perfiles de movimiento), es buena idea representarlos en forma gráfica. Esto puede lograrse trazando un movimiento como la posición o velocidad con respecto al tiempo.



**Figura 4.19** Perfil de movimiento trapezoidal

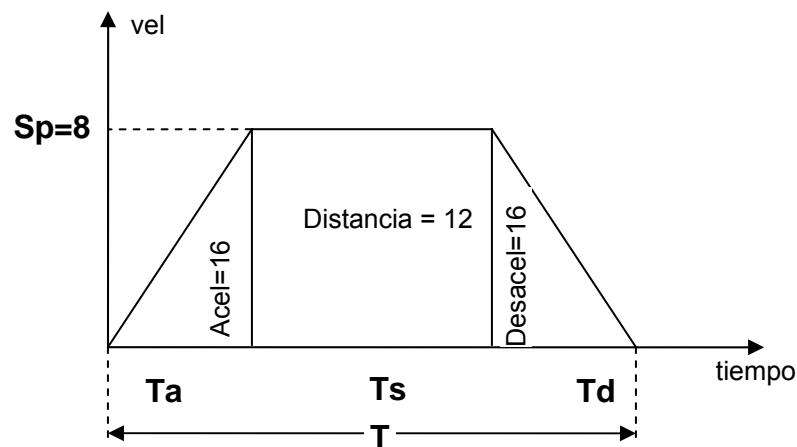
En la figura 4.19, se muestra el perfil de velocidad en movimiento trapezoidal, existen tres regiones en la gráfica: La aceleración, la parte de velocidad constante (slew speed) y la parte de desaceleración. A continuación se presentaran algunas propiedades del movimiento trapezoidal.

- T es el tiempo total del movimiento, Ta es el tiempo de aceleración, Ts es el tiempo de velocidad constante y Td es el tiempo de desaceleración.
- Sp es el valor de la velocidad constante.
- La razón de aceleración es la línea pendiente durante la parte de aceleración del perfil de velocidad y puede ser calculada utilizando la expresión siguiente:  $Acel = Sp/Ta$ .
- Similarmente, la razón de desaceleración es la pendiente de la línea de velocidad durante la parte de desaceleración en la gráfica.  $Desacel = Sp/Td$ .
- La distancia recorrida es igual al área debajo de la línea de velocidad.

Conociendo la información anterior se puede definir un perfil de movimiento a través de palabras clave de programación: SPEED, ACCEL, DECEL y la distancia; por ejemplo:

SPEED = 8, ACCEL = 16, DECEL = 16, MOVER = 12, GO

El código anterior generará un movimiento relativo de 12 unidades, alcanzando una velocidad de 8 unidades/s, con una razón de aceleración y desaceleración de 16 unidades/s<sup>2</sup>.



**Figura 4.20** Perfil de movimiento trapezoidal

La razón de aceleración se obtiene por:  $Acel = Sp/Ta$  donde:  $Ta = Sp/Acel = 8/16 = 0.5$  seg. Similarmente, para  $Td = Sp/Desacel = 8/16 = 0.5$  seg.

Se puede observar que la forma de la parte de aceleración es un triángulo. Por lo tanto, la distancia recorrida es igual al área debajo de la curva de velocidad, de tal manera que la distancia recorrida durante la aceleración puede calcularse utilizando la ecuación del área del triángulo:

Area =  $\frac{1}{2} \times$  base  $\times$  altura, Esto es,  $Da = \frac{1}{2} \times Ta \times Sp$

De aquí que la distancia recorrida durante la aceleración es  $Da = \frac{1}{2} \times \frac{1}{2} \times 8 = 2$  unidades. Como la razón de desaceleración es la misma que la razón de aceleración, entonces su valor también es de 2 unidades. Se sabe que la distancia total recorrida es la suma de todas las distancias:  $D = Da + Ds + Dd$ , donde:  $Ds = D - Da - Dd = 8$  unidades. El área debajo de la sección de velocidad constante es un cuadrado,

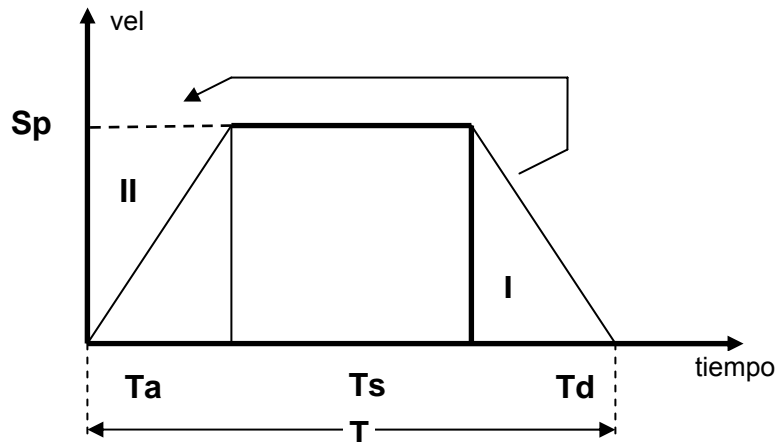
$Ds = Sp \times Ts$ , donde:  $Ts = Ds / Sp = 8/8 = 1$  s. El tiempo total de movimiento:  $T = Ta + Ts + Td = \frac{1}{2} + 1 + \frac{1}{2} = 2$  s

Es muy común que no toda la información necesaria para definir un movimiento se conozca. Por ejemplo, para un movimiento de 20 unidades en 600 ms. En esa situación con una gráfica Velocidad –Tiempo y haciendo ciertas observaciones se consigue más información.

En este ejemplo la velocidad constante y las razones de aceleración y desaceleración no han sido definidas. Parece que no está claro que valores usar, con la gráfica se puede asumir valores para la aceleración y desaceleración. Con un tiempo total de movimiento de 600 ms, se puede usar  $Ta = 200$  ms,  $Ts = 200$  y  $Td = 200$ . Se pueden utilizar libremente estos valores de tal forma que se simplifiquen los cálculos y se pueda evaluar la información desconocida.

En la figura 4.21, se muestra que el trapecio está compuesto por dos triángulos y un rectángulo. Si el tiempo de aceleración es igual al tiempo de desaceleración, entonces

sus áreas son iguales, se puede observar que el área I puede representar el área II y formar un rectángulo.



**Figura 4.21** El área del trapecio se transforma en el área de un rectángulo.

El área de un rectángulo  $A = axb$  o usando los parámetros de movimiento: Distancia =  $Sp \times (Ta + Ts)$ .

Rearreglando, se obtiene;  $Sp = \text{Distancia} / (Ta + Ts)$  **(4.7)**

Como se conoce la velocidad y el tiempo de aceleración, se tiene;

$$\text{Acel} = Sp / Ta \quad \textbf{(4.8)}$$

En el ejemplo anterior un movimiento de 20 unidades debe completarse en 600 ms. No se conocen los valores de SPEED, ACCEL y DECEL, pero se puede asumir que los tiempos de Accel, Velocidad constante y Decel son de 200 ms.

Usando la ecuación (4.7),  $Sp = \text{Distancia} / (Ta + Ts) = 12 / (0.2 + 0.2) = 30$  unidades/s.

Usando la ecuación (4.8),  $\text{Accel} = Sp / Ta = 30 / 0.2 = 150$  unidades/s<sup>2</sup>. En términos de Código de programación:

SPEED = 30, ACCEL = 150, DECEL = 150, MOVER = 20, GO

En conclusión, las únicas fórmulas necesarias para calcular en forma simple un movimiento, además de la gráfica de velocidad-tiempo es la del rectángulo y la pendiente de una recta.

#### IV.13 EL SISTEMA DE VISIÓN

El sistema de visión, es un sistema de procesamiento por computadora de imágenes, y se pretende realizar el seguimiento de objetos por color. Este sistema de visión trabaja a una velocidad de 30 Hz, con una resolución de 160x120 píxeles. El sistema también

es capaz de trabajar a una resolución más baja, 24 bits en formato RGB a 64x48 píxeles de resolución. La entrada de video es del tipo NTSC, dándole gran flexibilidad y compatibilidad con los sistemas actuales en el mercado. La tarjeta genera señales de salida a 60 Hz, en el mismo tipo NTSC. El sistema de visión se conecta a una computadora PC para monitorear las imágenes captadas.

#### IV.13.1 MODO DE USO

El sistema de visión permite algunos modos de uso:

- Utilizando algoritmos preprogramados el sistema puede funcionar enviando las señales a través del puerto serial a una computadora.
- El sistema de visión se puede programar utilizando el lenguaje de programación C<sup>®</sup> para controlar varios actuadores. Y este código puede ser llamado después de cada imagen capturada, se puede procesar la información visual, realizar binarización, segmentación etc.
- El sistema contiene un puerto entradas y salidas digitales, para conectarse a otros dispositivos, y dos puertos seriales, uno asíncrono y el otro síncrono. Además, el sistema soporta la programación de algoritmos de usuario, los cuales se alojan en la memoria de la tarjeta.

#### IV.13.2 COMPONENTES DEL SISTEMA DE VISIÓN

El sistema está integrado de dos tarjetas pequeñas de dimensiones 2.5" x 6.25" x 1.25" cuando se ensamblan. La parte posterior realiza el procesamiento de color y lo digitaliza de NTSC a 24 bits de RGB, luego envía las señales RGB a través de una tabla de búsqueda (Lookup Table).

La tarjeta puede se utiliza para reconocer tres colores diferentes al mismo tiempo. Los objetos son seguidos o reconocidos usando una tabla de color (Lookup Table). Múltiple objetos de color pueden ser seguidos a una velocidad de procesamiento de 60 Hz. Las operaciones del cálculo de orientación, pueden reducir el tiempo de operación. La resolución de trabajo se reduce a 24 bits.

La parte inferior de la tarjeta de procesamiento de imágenes es el procesador. Es un microcontrolador Motorola<sup>®</sup> 68332 con comunicación a una PC. Contiene una memoria de 256K de RAM (Expandible a 1MB sobre la tarjeta).

El microcontrolador 68332 es el único que contiene un coprocesador temporal, el cual se utiliza solo para el procesamiento de binarización. El microcontrolador contiene puertos de entradas y salidas digitales, además de las que se conectan al sistema de visión, contiene también un bus que se selecciona vía software [4.9]. Algunas de las aplicaciones para robots guiados por visión son: Manipulación de objetos o piezas,



ensamblaje, transportación, seguimiento de trayectorias, regulación, evasión de obstáculos, etc.

Es importante mencionar, que la mayoría de los robots que cuentan con sistemas de visión en la industria, no son aprovechados al máximo, debido a que la visión artificial solo se utiliza para localizar objetos en el espacio de trabajo del robot, aunque cabe señalar que los sistemas de visión aislados tienen mucha demanda en el control de calidad. La información visual resulta de mayor valor cuando se incorpora al lazo de control, o cuando se utiliza para obtener información cinemática de piezas u objetos en los procesos de fabricación en la industria.

La retroalimentación visual directa se compone de varias disciplinas, como son: La visión artificial, teoría de control, robótica y computación. La retroalimentación visual consiste en incorporar la información visual en el lazo de control. Esta información visual es obtenida a través del sistema de visión para generar una señal de error definida directamente en términos de un conjunto de características distintivas de imágenes (control basado en imágenes). Esta variable de error es utilizada con la información de la posición y velocidad articular por un controlador para cumplir con ciertos objetivos de control [4.10].

El problema de control de posición de robot manipuladores data de inicios de 1967, la compañía Unimation creó su primer robot en 1961. Siendo M. Whicman de “Stanford Artificial Intelligence” el primero en utilizar una cámara en un robot para localizar cajas blancas en un fondo negro.

Existen dos estrategias de control de posición de robot manipuladores con retroalimentación visual directa:

**IV.13.3 CÁMARA EN MANO:** Consiste en colocar la cámara de video en uno de los eslabones del robot manipulador, capturando imágenes del medio ambiente. El control determina los pares que se le aplican a las articulaciones del robot manipulador, para que el efector final describa la trayectoria deseada sobre el plano de la imagen. La tarea más simple es cuando el objeto no tiene movimiento y la trayectoria deseada es una posición o punto en el plano de la imagen [4.11].

**IV.13.4 CÁMARA FIJA:** Consiste en colocar la cámara de video fija con respecto al sistema de coordenadas fijo en el robot, capturando imágenes del robot y de su medio ambiente. El problema de control es determinar los pares que se deben aplicar a las articulaciones del robot, para que el efector final alcance la posición deseada en el plano de la imagen [4.12].

Es importante señalar, que en este trabajo se utilizará la estrategia cámara fija, debido a que solo es necesario conocer la posición y la distancia entre las piezas, y esta estrategia satisface el objetivo.

Ahora bien, para resolver el problema de control se han planteado dos esquemas:

*“Mira y Mueve”*: La primera fase *“mira”*, opera como un lazo externo utilizando la información visual proveniente de la cámara para encontrar la posición del objeto en coordenadas articulares. La segunda fase, *“mueve”* consiste en emplear las posiciones deseadas determinadas en la primera fase como referencias para un controlador en coordenadas articulares.

La información visual, entonces, es utilizada en lazo abierto, ya que no existe retroalimentación, ni del estado, ni de las salidas del robot. Cabe señalar, que en nuestros días se está trabajando con las herramientas y elementos de la teoría de control y la retroalimentación visual para el diseño de controladores [4.13,4.14,4.15,4.16].

Hasta la fecha, los primeros esquemas de control en la configuración cámara fija que contienen un análisis riguroso de estabilidad incluyendo la dinámica no lineal del robot manipulador son los presentados en [4.17,4.18], mientras que para sistemas robóticos cámara en mano se encuentran los trabajos [4.13,4.14].

Sin embargo, en la literatura científica del área de retroalimentación visual (visual servoing), se ha trabajado mucho para resolver el problema de posición utilizando la información visual [4.19], Aquí se considera a la dinámica no-lineal sin efecto en el lazo de retroalimentación visual, considerando al robot un sistema ideal de posicionamiento.

El sistema visión está relacionado con un conjunto de parámetros denominados parámetros extrínsecos e intrínsecos que intervienen de manera importante en la estructura del modelo matemático no-lineal del sistema de visión

#### IV.14 DESCRIPCIÓN DEL SISTEMA ROBÓTICO

La cámara de video es considerada como un elemento sensor que permite medir posición y orientación de objetos visibles dentro de su campo de visión [4.20]. La descripción del modelo de visión requiere un conjunto de sistemas de referencia cartesianos, se utilizará la regla de la mano derecha.

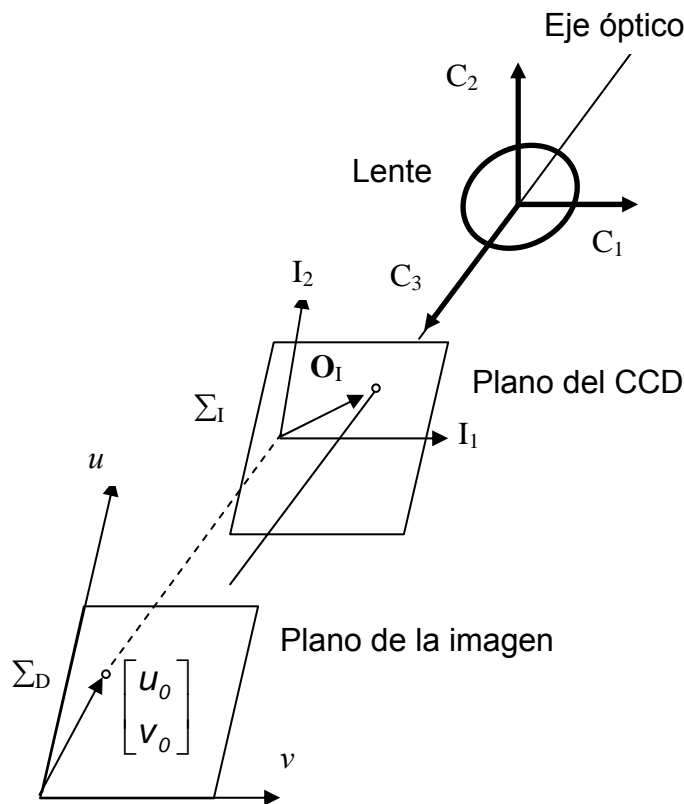
Se plantea la notación y la terminología necesaria para la descripción del sistema. Sea  $\Sigma_R = \{R_1, R_2, R_3\}$  un sistema de referencia cartesiano asociado al espacio de trabajo del robot manipulador, cuyo origen se encuentra localizado en la base del mismo. La orientación de los ejes principales  $R_1, R_2, R_3$  se selecciona de manera conveniente a la estructura del robot. Las coordenadas de un punto  $X$  con respecto a  $\Sigma_R = \{R_1, R_2, R_3\}$  se presenta por  $X_R = [X_{R_1}, X_{R_2}, X_{R_3}]$ .

La cámara de video tiene asociada un sistema de referencia  $\Sigma_C = \{C_1, C_2, C_3\}$ , cuyo origen está definido en la intersección del eje óptico con el centro de la lente ver figura 4.16, Los ejes  $C_1$  y  $C_2$  definen un plano perpendicular al eje óptico, esto es, el eje  $C_3$  está alineado con el eje óptico. Un punto  $X$  con respecto al sistema de referencia de la cámara se denota por  $X_C = [X_{C_1} X_{C_2} X_{C_3}]^T$ , La localización relativa entre los sistemas de

referencia del robot  $\Sigma_R$  y el sistema de referencia de la cámara  $\Sigma_C$  está representado por el vector  $O_R^C$  por lo que el plano  $C_1$ - $C_2$  está localizado a una distancia  $O_{R_3}^C$  del plano  $R_1$ - $R_2$ .

El plano donde se proyecta la imagen óptica de la escena corresponde al arreglo discreto de elementos fotosensibles CCD, el cual tiene asociado un sistema de referencia bidimensional  $\Sigma_I = \{I_1, I_2\}$ , cuyo origen se encuentra localizado en el centro geométrico del arreglo CCD. La posición de la intersección del eje óptico con  $\Sigma_I$  es denotado  $O_I$ , como se muestra en la figura 4.16, El vector  $O_I$ , puede ser diferente a cero debido a defectos internos de ensamblaje entre los lentes y el arreglo CCD. Un punto  $X$  en el plano de la imagen está representado por  $X_I = [X_{I_1} \ X_{I_2}]^T$ ; los ejes  $I_1$  e  $I_2$  son paralelos y apuntan en la misma dirección de los ejes  $C_1$  y  $C_2$ , respectivamente. El eje  $C_3$  apunta hacia el plano del CCD, de tal forma que los puntos  $X_C$  que están en frente de la cámara tienen la componente  $X_{C_3}$ , negativa.

La imagen de la escena sobre el arreglo CCD es digitalizada y transferida hacia la pantalla de la computadora para su exhibición. Sobre la pantalla del monitor se define un sistema de coordenadas bidimensional denotado por  $\Sigma_D = \{u, v\}$ , cuyo origen se encuentra localizado en la esquina superior izquierda del monitor con los ejes  $u$  y  $v$  paralelos a los renglones y columnas respectivamente. El vector  $[u_0 \ v_0]^T$  representa el centro de la imagen. Las coordenadas de un punto sobre la pantalla del monitor son denotadas por  $X_D = [u \ v]^T$  y tienen unidades de *píxeles* [4.21].



**Figura 4.22** Sistema de coordenadas de referencia

Se consideran las siguientes hipótesis:

1.- Los ejes  $C_3$  y  $R_3$  son paralelos y apuntan en la misma dirección, es decir, los planos  $R_1$ - $R_2$  y  $C_1$ - $C_2$  son paralelos. También se ha supuesto que existe un ángulo de rotación  $\theta$  alrededor del eje  $R_3$  para denotar la orientación relativa del sistema de referencia  $\Sigma_C$  con respecto al sistema de referencia  $\Sigma_R$  [4.22], como se muestra en la figura 4.22.

2.- El plano del CCD es perpendicular al eje óptico, es decir, los planos  $C_1$ - $C_2$  y  $I_1$ - $I_2$  son paralelos. Además, la imagen de un punto  $X_C$  con respecto a la cámara está bien enfocada en el plano del arreglo CCD, lo que significa que el plano del arreglo CCD se encuentra a una distancia  $|\lambda/(\lambda+XC_3)|$  del plano  $C_1$ - $C_2$ , donde  $\lambda > 0$  es la distancia focal [4.23].

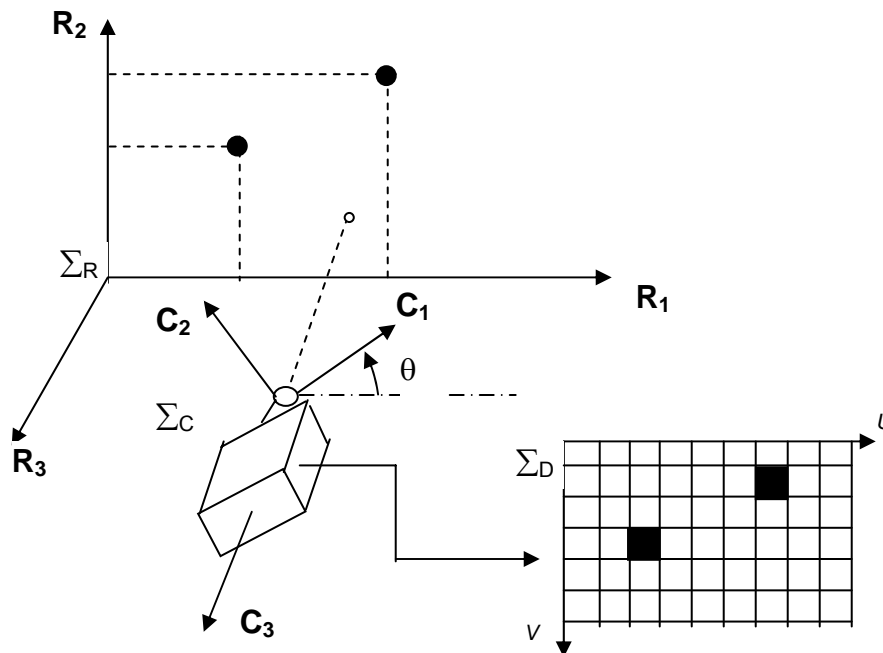


Figura 4.23 Sistema de visión y sus coordenadas.

#### IV.15 MODELO DEL SISTEMA DE VISIÓN

Las ecuaciones que determinan un punto  $X_R$  en el espacio sobre un punto en la pantalla del monitor, denotado por  $X_D = [u \ v]^T$ , constituyen el modelo del sistema de visión, el cual está formado por un conjunto de transformaciones que a continuación se describen.

#### IV.15.1 TRANSFORMACIÓN DE COORDENADAS CARTESIANAS A COORDENADAS DE LA CÁMARA

La transformación de coordenadas  $\mathbf{X}_R$  a coordenadas de la cámara  $\mathbf{X}_C$  representa una transformación de cuerpo rígido que se obtienen a partir de la siguiente ecuación (4.9) [4.24].

$$\mathbf{X}_C = \mathbf{R}_R^{C^T} [\mathbf{X}_R - \mathbf{O}_R^C] \quad (4.9)$$

donde  $\mathbf{R}_R^C \in \text{SO}(3)$  (SO es el grupo especial ortogonal) es una matriz de rotación que representa la orientación del sistema de la cámara  $\Sigma_C$  con respecto al sistema de referencia  $\Sigma_R$ , y de acuerdo a la orientación de la cámara con respecto al sistema del robot  $\Sigma_R$  descrita en la hipótesis 1, la matriz  $\mathbf{R}_R^C$  está dada por [4.25,4.26,4.27]:

$$\mathbf{R}_R^C(\theta) = \begin{bmatrix} \cos(\theta) & -\text{sen}(\theta) & 0 \\ \text{sen}(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.10)$$

#### IV.15.2 TRANSFORMACIÓN DE PERSPECTIVA

La transformación de perspectiva proyecta las coordenadas de la cámara sobre las coordenadas en el plano CCD. Usando el modelo de óptica geométrica para lentes delgadas y tomando en cuenta el efecto de distorsión radial, la proyección de un punto  $\mathbf{X}_C$  sobre un punto  $\mathbf{X}_I$  del plano CCD está dada por la siguiente expresión:

$$\mathbf{X}_I = \left[ 1 + K_1 \frac{\lambda^2}{(\lambda + X_{C_3})^2} \left\| \begin{bmatrix} X_{C_1} \\ X_{C_2} \end{bmatrix} \right\|^2 \right] \frac{\lambda}{\lambda + X_{C_3}} \begin{bmatrix} X_{C_1} \\ X_{C_2} \end{bmatrix} + \mathbf{O}_I \quad (4.11)$$

donde  $K_1$  es el coeficiente de distorsión radial. La ecuación (4.11) incluye el fenómeno de distorsión radial de la lente. Su efecto es causar un desplazamiento interior o exterior de un punto de la imagen desde su posición ideal. Un desplazamiento radial negativo de los puntos de la imagen es llamado distorsión barril, esto causa que los puntos se junten (se apiñonen) y dá la apariencia que la escala se decrementa. Un desplazamiento radial positivo es conocido como distorsión de corsé y su efecto causa que los puntos se extiendan e incremente la escala. La distorsión radial es estrictamente simétrica alrededor del eje óptico [4.25,4.26,4.27].

La distancia focal efectiva  $\lambda/(\lambda+X_{C_3})$  en el modelo (4.12) es negativa debido a que ambos sistemas  $\Sigma_R$  y  $\Sigma_C$  están definidos por la regla de la mano derecha. Obsérvese que sólo en el caso  $|X_{C_3}| \gg \lambda$  se tiene  $\lambda/(\lambda+X_{C_3}) \approx \lambda/X_{C_3}$ .

En la transformación de coordenadas del plano del CCD a coordenadas de la imagen, la posición de un punto  $\mathbf{X}_I$  en el plano del CCD es transformada hacia un punto  $\mathbf{X}_D$  sobre el plano de la imagen a través de la siguiente transformación lineal [4.28].

$$\mathbf{x}_D = \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} -\alpha_u & 0 \\ 0 & \alpha_v \end{bmatrix} \mathbf{x}_I + \begin{bmatrix} U_0 \\ V_0 \end{bmatrix} \quad (4.12)$$

donde  $\alpha_u > 0$ ,  $\alpha_v > 0$  representan los factores de escala en píxeles/m a lo largo de los ejes  $I_1$  e  $I_2$  del plano CCD respectivamente y  $U_0$ ,  $V_0$  denotan la posición en píxeles del origen del sistema  $\Sigma_I$  con respecto al sistema  $\Sigma_D$ .

### IIV.15.3 MODELO COMPLETO DEL SISTEMA DE VISIÓN

Las transformaciones anteriores caracterizan al modelo del sistema de visión, que en forma completa tienen la siguiente estructura:

$$\begin{bmatrix} \mathbf{Xc}_1 \\ \mathbf{Xc}_2 \\ \mathbf{Xc}_3 \end{bmatrix} = \mathbf{R}_R^C(\theta)^T [\mathbf{x}_R - \mathbf{o}_R^C]$$

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \frac{\lambda}{\lambda + \mathbf{Xc}_3} + K_1 \frac{\lambda^3}{(\lambda + \mathbf{Xc}_3)^3} \|\mathbf{Xc}_1\|^2 \\ \frac{\lambda^3}{(\lambda + \mathbf{Xc}_3)^3} \|\mathbf{Xc}_2\|^2 \end{bmatrix} \begin{bmatrix} -\alpha_u & 0 \\ 0 & \alpha_v \end{bmatrix} \begin{bmatrix} \mathbf{Xc}_1 \\ \mathbf{Xc}_2 \end{bmatrix} + \begin{bmatrix} -\alpha_u & 0 \\ 0 & \alpha_v \end{bmatrix} \mathbf{o}_I + \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} \quad (4.13)$$

Los parámetros extrínsecos están asociados a la posición y orientación de la cámara con respecto al sistema  $\Sigma_R$ , tales como  $\theta$  y  $\mathbf{o}_R^C$ . Mientras que los parámetros intrínsecos se refieren a los aspectos ópticos y geométricos de la cámara, por ejemplo,  $\alpha_u$ ,  $\alpha_v$ ,  $\lambda$  y  $\mathbf{o}_I$ . La figura 4.24, describe el diagrama de bloques del modelo de visión:

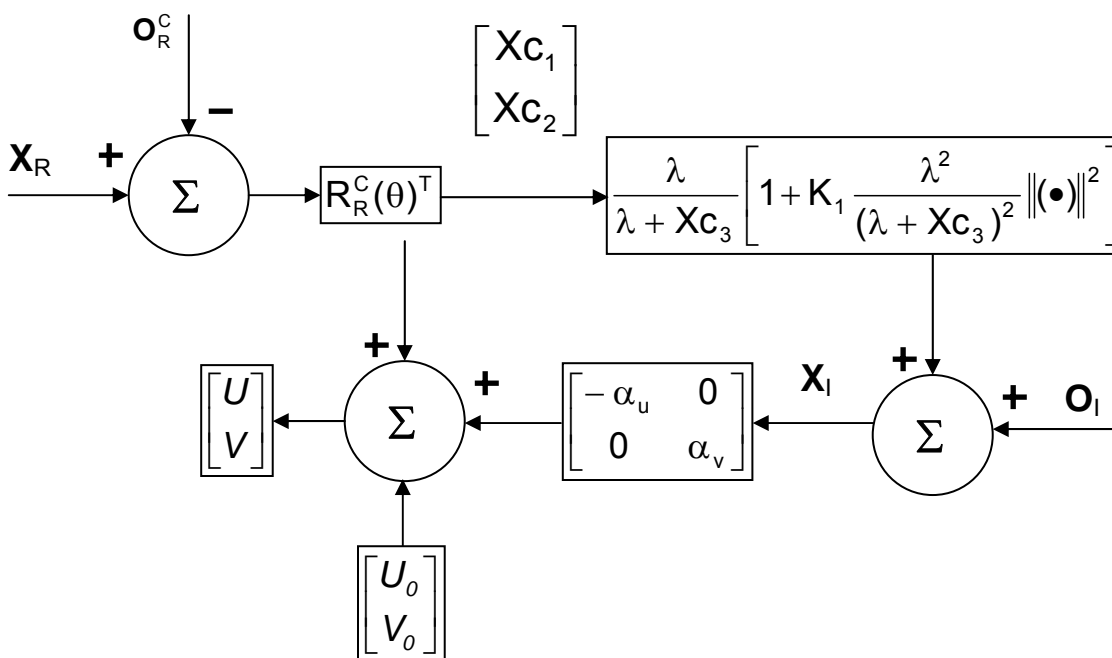


Figura 4.24 Diagrama de bloques del modelo de visión.

#### IV.15.4 PARAMETRIZACIÓN LINEAL DEL SISTEMA DE VISIÓN

El modelo del sistema de visión (4.11) es una función no lineal de los parámetros extrínsecos e intrínsecos. El reconocimiento de que ese modelo puede ser expresado como una regresión lineal en términos de un conjunto adecuado de parámetros extrínsecos e intrínsecos, i.e.

La propiedad de linealidad del modelo (4.13) lo hace atractivo en aplicaciones de control de robots manipuladores con retroalimentación visual y calibración de sistema de visión, que en este caso es el objetivo de este tema la calibración del sistema de visión [4.25].

**Teorema 4.1.** El sistema de visión caracterizado por el modelo (4.13) es lineal en términos de un conjunto adecuado de parámetros extrínsecos e intrínsecos [4.25], i.e.

$$\begin{bmatrix} u \\ v \end{bmatrix} = \psi(X_{R_1}, X_{R_2})^T \rho(Xc_3) \quad (4.14)$$

donde  $\psi(X_{R_1}, X_{R_2}) \in \mathfrak{R}^{16 \times 2}$  representa la matriz de regresión de funciones conocidas  $X_{R_1}$  y  $X_{R_2}$  y  $\hat{\rho} \in \mathfrak{R}^{16}$  es el vector de parámetros dependiente de constantes extrínsecas e intrínsecas y de la profundidad del punto, ver prueba en [4.29].

La selección de los elementos de la matriz de regresión  $\psi(X_{R_1}, X_{R_2})$  están dadas como:

$$\begin{aligned} \Psi_{1,1} &= -\Psi_{4,2} = X_{R_1}, \\ \Psi_{2,1} &= \Psi_{3,2} = X_{R_2}, \\ \Psi_{5,1} &= \Psi_{6,2} = 1, \\ \Psi_{7,1} &= -\Psi_{10,2} = X_{R_1} [X_{R_1}^2 + X_{R_2}^2], \\ \Psi_{8,1} &= \Psi_{9,2} = X_{R_2} [X_{R_1}^2 + X_{R_2}^2], \\ \Psi_{11,1} &= -\Psi_{14,2} = X_{R_1}^2, \\ \Psi_{12,1} &= -\Psi_{13,2} = X_{R_1} X_{R_2}, \\ \Psi_{15,1} &= \Psi_{16,2} = X_{R_2}^2 \end{aligned} \quad (4.15)$$

mientras que los restantes elementos tienen valor cero. Los componentes correspondientes del vector de parámetros  $\hat{\rho}$  están listados en la tabla 4.5:

La complejidad del modelo es reducida en ausencia de distorsión óptica; en este caso se tiene el siguiente resultado:

**Colorario 4.1.** Considere el sistema de visión (4.13) libre de distorsión, es decir,  $K_1 = 0$ , entonces;

$$\begin{bmatrix} u \\ v \end{bmatrix} = \psi_L(X_{R_1}, X_{R_2})^T \rho_L \quad (4.16)$$

donde

$$\Psi_L(X_{R_1}, X_{R_2}) = \begin{bmatrix} X_{R_1} & X_{R_2} & 0 & 0 & 1 & 0 \\ 0 & 0 & X_{R_2} & -X_{R_1} & 0 & 1 \end{bmatrix} \quad (4.17)$$

y los componentes del vector  $p_L \in \mathfrak{R}^6$  se encuentran definidos en la tabla 4.6.

Los componentes  $p_7$  a  $p_{16}$  se anulan cuando  $K_1 = 0$ , véase prueba en [4.29].

Tabla 4.5 Componentes del vector de parámetros	
$p_1$	$-\alpha_u \frac{\lambda}{\lambda + X_{C_3}} \left[ c\theta \left[ 1 + k_1 \frac{\lambda^2}{(\lambda + X_{C_3})^2} [\mathbf{o}_{R_1}^{C^2} + \mathbf{o}_{R_2}^{C^2}] \right] + 2k_1 \frac{\lambda^2}{(\lambda + X_{C_3})^2} \mathbf{o}_{R_1}^C [\mathbf{o}_{R_1}^C c\theta + \mathbf{o}_{R_2}^C s\theta] \right]$
$p_2$	$-\alpha_u \frac{\lambda}{\lambda + X_{C_3}} \left[ s\theta \left[ 1 + k_1 \frac{\lambda^2}{(\lambda + X_{C_3})^2} [\mathbf{o}_{R_1}^{C^2} + \mathbf{o}_{R_2}^{C^2}] \right] + 2k_1 \frac{\lambda^2}{(\lambda + X_{C_3})^2} \mathbf{o}_{R_1}^C [\mathbf{o}_{R_1}^C c\theta + \mathbf{o}_{R_2}^C s\theta] \right]$
$p_3$	$\alpha_v \frac{\lambda}{\lambda + X_{C_3}} \left[ c\theta \left[ 1 + k_1 \frac{\lambda^2}{(\lambda + X_{C_3})^2} [\mathbf{o}_{R_1}^{C^2} + \mathbf{o}_{R_2}^{C^2}] \right] + 2k_1 \frac{\lambda^2}{(\lambda + X_{C_3})^2} \mathbf{o}_{R_2}^C [\mathbf{o}_{R_2}^C c\theta - \mathbf{o}_{R_1}^C s\theta] \right]$
$p_4$	$\alpha_v \frac{\lambda}{\lambda + X_{C_3}} \left[ s\theta \left[ 1 + k_1 \frac{\lambda^2}{(\lambda + X_{C_3})^2} [\mathbf{o}_{R_1}^{C^2} + \mathbf{o}_{R_2}^{C^2}] \right] - 2k_1 \frac{\lambda^2}{(\lambda + X_{C_3})^2} \mathbf{o}_{R_1}^C [\mathbf{o}_{R_2}^C c\theta - \mathbf{o}_{R_1}^C s\theta] \right]$
$p_5$	$\alpha_u \frac{\lambda}{\lambda + X_{C_3}} \left[ 1 + k_1 \frac{\lambda^2}{(\lambda + X_{C_3})^2} [\mathbf{o}_{R_1}^{C^2} + \mathbf{o}_{R_2}^{C^2}] \right] [\mathbf{o}_{R_1}^C c\theta + \mathbf{o}_{R_2}^C s\theta] - \alpha_u \mathbf{o}_{l_1} + u_0$
$p_6$	$-\alpha_v \frac{\lambda}{\lambda + X_{C_3}} \left[ 1 + k_1 \frac{\lambda^2}{(\lambda + X_{C_3})^2} [\mathbf{o}_{R_1}^{C^2} + \mathbf{o}_{R_2}^{C^2}] \right] [\mathbf{o}_{R_2}^C c\theta - \mathbf{o}_{R_1}^C s\theta] + \alpha_v \mathbf{o}_{l_2} + v_0$
$p_7$	$-k_1 \alpha_u \frac{\lambda^3}{(\lambda + X_{C_3})^3} c\theta$
$p_8$	$-k_1 \alpha_u \frac{\lambda^3}{(\lambda + X_{C_3})^3} s\theta$
$p_9$	$k_1 \alpha_v \frac{\lambda^3}{(\lambda + X_{C_3})^3} c\theta$
$p_{10}$	$k_1 \alpha_v \frac{\lambda^3}{(\lambda + X_{C_3})^3} s\theta$
$p_{11}$	$k_1 \alpha_u \frac{\lambda^3}{(\lambda + X_{C_3})^3} [3\mathbf{o}_{R_1}^C c\theta + \mathbf{o}_{R_2}^C s\theta]$
$p_{12}$	$k_1 2\alpha_u \frac{\lambda^3}{(\lambda + X_{C_3})^3} [\mathbf{o}_{R_1}^C s\theta + \mathbf{o}_{R_2}^C c\theta]$



p <sub>13</sub>	$-k_1 2\alpha_v \frac{\lambda^3}{(\lambda + X_{C_3})^3} [\mathbf{O}_{R_1}^C c\theta - \mathbf{O}_{R_2}^C s\theta]$
p <sub>14</sub>	$-k_1 \alpha_v \frac{\lambda^3}{(\lambda + X_{C_3})^3} [3\mathbf{O}_{R_1}^C s\theta - \mathbf{O}_{R_2}^C c\theta]$
p <sub>15</sub>	$k_1 \alpha_u \frac{\lambda^3}{(\lambda + X_{C_3})^3} [3\mathbf{O}_{R_2}^C s\theta + \mathbf{O}_{R_1}^C c\theta]$
p <sub>16</sub>	$-k_1 \alpha_v \frac{\lambda^3}{(\lambda + X_{C_3})^3} [3\mathbf{O}_{R_2}^C c\theta - \mathbf{O}_{R_1}^C s\theta]$

Tabla 4.6 Vector de parámetros (Sistema de visión libre de distorsión)	
P <sub>L1</sub>	$-\alpha_u \frac{\lambda}{\lambda + X_{C_3}} c\theta$
P <sub>L2</sub>	$-\alpha_u \frac{\lambda}{\lambda + X_{C_3}} s\theta$
P <sub>L3</sub>	$-\alpha_v \frac{\lambda}{\lambda + X_{C_3}} c\theta$
P <sub>L4</sub>	$-\alpha_v \frac{\lambda}{\lambda + X_{C_3}} s\theta$
P <sub>L5</sub>	$\alpha_u \frac{\lambda}{\lambda + X_{C_3}} [\mathbf{O}_{R_1}^C c\theta + \mathbf{O}_{R_2}^C s\theta] - \alpha_u \mathbf{o}_{11} + u_0$
P <sub>L6</sub>	$-\alpha_v \frac{\lambda}{\lambda + X_{C_3}} [\mathbf{O}_{R_2}^C c\theta - \mathbf{O}_{R_1}^C s\theta] - \alpha_v \mathbf{o}_{12} + v_0$

#### IV.16 CALIBRACIÓN DEL SISTEMA DE VISIÓN

El modelo de visión definido por la ecuación (4.13) depende de manera explícita de los parámetros extrínsecos e intrínsecos, es decir;

$$\begin{bmatrix} u \\ v \end{bmatrix} = \mathbf{f}_D(\mathbf{X}_R; u_c, v_c, \alpha_u, \alpha_v, \mathbf{O}_1, K_1, \lambda, \theta, \mathbf{O}_R^C) \quad (4.18)$$

El problema de calibración para sistemas de visión consiste en determinar los parámetros extrínsecos e intrínsecos usando un conjunto de puntos conocidos en coordenadas del sistema  $\Sigma_R$  y sus correspondientes posiciones observadas en el sistema de referencia de  $\Sigma_D$ . Existe una literatura extensa sobre técnicas de calibración para sistemas de visión que de manera general pueden ser clasificadas dentro de las siguientes categorías:

**IV.16.1 OPTIMIZACIÓN LINEAL:** Esta clase de técnicas emplean algoritmos iterativos que minimizan errores residuales de las ecuaciones que contienen los parámetros de la cámara; la mayoría de las técnicas de calibración que pertenecen a esta categoría se emplean en “*fotogrametría*” [4.30], se define como el área relacionada con la obtención de información de un objeto físico a través de imágenes [4.10]. Una desventaja de este tipo de algoritmos es que pueden producir una solución errónea a no ser que se disponga de una condición inicial adecuada. Además, si los parámetros de distorsión son incluidos, la minimización puede ser inestable si el procedimiento de iteraciones no está bien condicionado [4.26].

**IV.16.2 SOLUCIÓN EN FORMA CERRADA:** En esta clase de esquemas, los parámetros del sistema de visión son estimados a través de algoritmos no iterativos basados en una solución en forma cerrada [4.31]. Un conjunto de parámetros auxiliares son definidos en términos de los parámetros extrínsecos e intrínsecos. A partir de los parámetros auxiliares, los parámetros de interés son obtenidos resolviendo un conjunto de ecuaciones lineales. La ventaja de este tipo de algoritmos es que son rápidos y presenta como desventaja el no incluir el modelo de distorsión [4.32].

**IV.16.3 MÉTODO EN DOS PASOS:** Consisten en descomponer los parámetros de calibración en dos grupos. El primero está formado por parámetros extrínsecos, es decir, de translación y rotación relativos al sistema de referencia del robot. La obtención de estos parámetros consiste en registrar un número de puntos (mayor a cinco) y establecer un sistema sobredeterminado de ecuaciones lineales, el cual es resuelto a través de mínimos cuadrados. El segundo grupo de parámetros es obtenido empleando esquemas de optimización (paso descendente, gradiente, etc.). La solución obtenida en el primer paso es empleada como condición inicial para encontrar  $\lambda$ ,  $\mathbf{O}_1$  y  $K_1$  [4.33]. Una aplicación inmediata de la propiedad de linealidad expresada por (4.14), es que permite obtener un algoritmo de calibración para sistemas de visión, el cual puede ser clasificado en la categoría de técnicas con solución cerrada. Sin embargo, el nuevo algoritmo de calibración sí incluye el fenómeno de distorsión radial.

**Colorario 4.2.** La propiedad de linealidad del sistema de visión (4.13) permite obtener expresiones en forma cerrada para los parámetros extrínsecos e intrínsecos de acuerdo a las siguientes expresiones:

$$\theta = \text{atan}\left(\frac{p_8}{p_7}\right) \quad (4.19)$$

$$\frac{\alpha_u}{\alpha_v} = -\left(\frac{p_7}{p_9}\right) \quad (4.20)$$

$$\mathbf{O}_{R_1}^C = \frac{-p_{11} + \frac{1}{2} \tan(\theta) p_{12}}{p_7 [3 - \tan(\theta)^2]} \quad (4.21)$$

$$\mathbf{O}_{R_2}^C = \frac{p_{11} + \tan(\theta) - \frac{3}{2}p_{12}}{p_7[3 - \tan(\theta)^2]} \quad (4.22)$$

$$-\alpha_u \mathbf{O}_{I_2} + u_0 = p_5 - \left( \text{aux}_u - \frac{p_7}{\cos(\theta)} (\mathbf{O}_{R_1}^{C^2} + \mathbf{O}_{R_2}^{C^2}) \right) (\mathbf{O}_{R_1}^C \cos(\theta) + \mathbf{O}_{R_2}^C \sin(\theta)) \quad (4.23)$$

$$\alpha_v \mathbf{O}_{I_2} + v_0 = p_6 + \left( \text{aux}_v + \frac{p_9}{\cos(\theta)} (\mathbf{O}_{R_1}^{C^2} + \mathbf{O}_{R_2}^{C^2}) \right) (\mathbf{O}_{R_2}^C \cos(\theta) - \mathbf{O}_{R_1}^C \sin(\theta)) \quad (4.24)$$

donde

$$\text{aux}_u = -\left( \frac{p_1}{\cos(\theta)} - \frac{p_7}{\cos(\theta)} (\mathbf{O}_{R_1}^{C^2} + \mathbf{O}_{R_2}^{C^2}) - 2 \frac{p_7}{\cos(\theta)^2} \mathbf{O}_{R_1}^C (\mathbf{O}_{R_1}^C \cos(\theta) + \mathbf{O}_{R_2}^C \sin(\theta)) \right)$$

$$\text{aux}_v = \left( \frac{p_3}{\cos(\theta)} - \frac{p_9}{\cos(\theta)} (\mathbf{O}_{R_1}^{C^2} + \mathbf{O}_{R_2}^{C^2}) - 2 \frac{p_9}{\cos(\theta)^2} \mathbf{O}_{R_2}^C (\mathbf{O}_{R_2}^C \cos(\theta) - \mathbf{O}_{R_1}^C \sin(\theta)) \right)$$

Existen algunas restricciones impuestas sobre el ángulo de rotación  $\theta$ , tales como el ángulo de rotación  $\theta \neq (2n+1)\pi/2$ .

El proceso de calibración generalmente es realizado a través de una plantilla que consiste de un arreglo rectangular de cuadrados, diamantes o círculos cuyas coordenadas son conocidas en el sistema de referencia  $\Sigma_R$  [4.26,4.33, 4.31].

En este trabajo se ha realizado la calibración del sistema de visión a través de la plantilla. Para esta finalidad en el procedimiento de calibración se han considerado las siguientes suposiciones:

- La plantilla tiene asociado un sistema de referencia  $\Sigma_M = \{M_1, M_2, M_3\}$  donde los ejes  $M_3$  y  $R_3$  son paralelos y apuntan en la misma dirección. Es decir, los planos  $R_1$ - $R_2$  y  $M_1$ - $M_2$  Son paralelos. Además, existe un ángulo de rotación  $\beta_M$  con valor conocido alrededor del eje  $R_3$  para denotar la orientación relativa del sistema de referencia  $\Sigma_M$  con respecto al sistema de referencia  $\Sigma_R$ , por convención el sentido positivo de  $\beta_M$  está dado por la regla de la mano derecha. El origen del sistema de referencia  $\Sigma_M$  con respecto a  $\Sigma_R$  está representado por  $\mathbf{O}_R^M$ .
- La transformación de coordenadas de la plantilla  $\Sigma_M$  a coordenadas  $\Sigma_R$  está dada por:

$$X_R = \mathbf{O}_R^M + R_R^M(\beta_M) X_M \quad (4.25)$$

donde  $X_M$  representa las coordenadas de cada punto en  $\Sigma_M$  y  $R_R^M(\beta_M) \in SO(3)$  es la matriz de rotación del sistema de referencia  $\Sigma_M$  con respecto al sistema de referencia  $\Sigma_R$ :

$$R_R^M(\beta_M) = \begin{bmatrix} \cos(\beta_M) & -\text{sen}(\beta_M) & 0 \\ \text{sen}(\beta_M) & \cos(\beta_M) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.26)$$

#### IV.17 RESULTADOS EXPERIMENTALES DE CALIBRACIÓN DEL SISTEMA DE VISIÓN

La calibración del sistema de visión fue realizada usando una plantilla fabricada en Autocad® e impresora láser. La plantilla de calibración consistió de un arreglo rectangular de 12x8 cuadrados; la distancia de centro a centro entre dos cuadrados fue de 2.25 cm y la longitud de cada cuadrado es de 1.2 cm. Los centros geométricos de cada cuadrado fueron considerados como los puntos de calibración y fueron las entradas en el modelo de regresión del sistema de visión, ver ecuación (4.18).

La plantilla fue instalada en el extremo final del robot y rotada con respecto al plano  $R_1$ - $R_2$  un ángulo  $\beta_M = q_1 + q_2 = 73.32^\circ$  alrededor del eje  $R_3$ , donde  $q_1 = 35^\circ$  y  $q_2 = 38.32^\circ$  son los desplazamientos angulares de las articulaciones del hombro y del codo respectivamente. Las coordenadas de la cámara de CCD con respecto a  $\Sigma_R$  fueron colocadas en  $O_R^C = [0.70 \ -0.48 \ 0.45]$  m. El plano de la lente fue colocado aproximadamente paralelo con el plano de la mascarilla. El sistema de referencia de la cámara  $\Sigma_M$  fue rotado un ángulo  $\theta = 20^\circ$  con respecto al sistema  $\Sigma_R$ . El ángulo  $\theta$  y el vector  $O_R^C$  forman parte de los parámetros extrínsecos del sistema de visión. Puesto que los puntos de calibración están sobre el plano  $M_1$ - $M_2$ , el vector  $O_R^C$  fue seleccionado como  $O_R^M = [X_{R_1}^M, X_{R_2}^M, X_{R_3}^M, 0]^T$ , por lo que  $X_{R_3} = X_{M_3} = 0$ .

A continuación se describe el procedimiento de calibración experimental.

- La transformación de la plantilla de calibración a coordenadas cartesianas está dada por:

$$X_R = \underbrace{\begin{bmatrix} l_1 \text{sen}(q_1) + l_2 \cos(q_1 + q_2) \\ -l_1 \cos(q_1) - l_2 \cos(q_1 + q_2) \\ 0 \end{bmatrix}}_{O_R^M} + \underbrace{\begin{bmatrix} \cos(q_1 + q_2) & -\text{sen}(q_1 + q_2) & 0 \\ \text{sen}(q_1 + q_2) & \cos(q_1 + q_2) & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{R_R^M(\beta_M)} X_M \quad (4.27)$$

donde  $l_1 = 0.250$  m y  $l_2 = 0.200$  m son las longitudes de las articulaciones del hombro y codo, respectivamente.

- Obtener la imagen binaria de la plantilla de calibración (el umbral de binarización puede ser obtenido por medio del histograma de la imagen en niveles de gris).
- Obtener las coordenadas de los centroides  $(u_i, v_i)$  para cada cuadrado  $i$ -ésimo en

la plantilla de calibración.

- Usando las coordenadas de los puntos de calibración y las coordenadas de los centroides, obtener la estimación de los parámetros  $\hat{\mathbf{p}}$ . Los parámetros del sistema de visión son obtenidos usando las ecuaciones (4.19)-(4.24)

Parámetro	Valor estimado	Unidades
$-\alpha_u O_{I_1} + u_0$	375.19	Píxeles
$\alpha_v O_{I_2} + v_0$	180.98	Píxeles
$O_{R_1}^C$	0.73	m
$O_{R_2}^C$	-0.48	m
$\theta$	16.89	° grados
$\frac{\alpha_u}{\alpha_v}$	1.24	

La tabla 4.7, muestra los parámetros obtenidos en forma experimental empleando el algoritmo del colorario 4.2. La norma del error de los centroides medidos por el sistema de visión y los centroides estimados por el modelo de regresión (4.18) empleando los parámetros  $\hat{\mathbf{p}}$  obtenidos en el proceso de calibración para los 96 puntos, se obtiene con:

$$\left\{ \left\| \begin{matrix} \tilde{u}(k) \\ \tilde{v}(k) \end{matrix} \right\| \right\}_{k=1}^{N=96} = \left\{ \left\| \begin{matrix} u(k) \\ v(k) \end{matrix} \right\| - \psi(\mathbf{X}_{R_1}, \mathbf{X}_{R_2})^T \hat{\mathbf{p}}(\mathbf{X}_{C_3}) \right\}_{k=1}^{N=96} \quad (4.28)$$

El promedio de la norma entre los centroides experimentales y los centroides estimados fue calculado como:

$$\frac{1}{N} \sum_{k=1}^{N=96} \left\| \begin{matrix} \tilde{u}(k) \\ \tilde{v}(k) \end{matrix} \right\| = 0.381 \text{ píxeles} \quad (4.29)$$

## IV.18 TAREA DEL SISTEMA DE VISIÓN

### IV.18.1 GENERALIDADES

La tarea principal del sistema de visión consiste en la identificación, localización y seguimiento de objetos en el espacio de trabajo del robot interceptor. El sistema de visión implantado en la integración robot-visión está desarrollado en el lenguaje de Matlab<sup>®</sup> 7.0 el cual cuenta con un toolbox de procesamiento de imágenes<sup>®</sup>. Además, puede capturar imágenes y realizar el procesamiento inmediatamente después de la captura. Las sub-tareas que tiene asignadas el sistema son **[4.34]**:

- Captura y preprocesamiento de la imagen (realizada por hardware)
- Segmentación (por programa)
- Reconocimiento (por programa)
- Etiquetado
- Obtención de las propiedades de la imagen (área, centroide y orientación)

### IV.18.2 CAPTURA Y PROCESAMIENTO DE LA IMAGEN [4.35].

La imagen empleada en el sistema integrado forma una matriz de 160x120 píxeles de resolución (codificados en formato RGB con 24 bits de profundidad), lo cual en términos de memoria es de 57600 bytes. Aunque las resoluciones pueden ser mayores, por ejemplo: 176x144, 160x120, 352x288 ó 640x480. Es importante considerar que la velocidad de procesamiento es crucial para nuestro objetivo, por un lado. Y por el otro, contar con un campo de visión de la cámara que permita “ver” si no todo, gran parte del espacio de trabajo del robot interceptor, obteniendo como consecuencia un ahorro de memoria con una pérdida mínima de información y una disminución del tiempo de procesamiento. Ya que la adquisición de la imagen se realiza a través del puerto USB<sup>®</sup>.

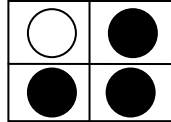
Durante el procesamiento se eliminan ruidos presentes en la imagen lo cual se realiza con una erosión con un elemento estructurante 4-conexo tal como se muestra en la figura 4.25 La erosión binaria es una operación morfológica que se define de la siguiente manera:

Definición 4.1. Sean A y B conjuntos en un espacio euclideo n-dimensional, entonces la erosión de A por B es el conjunto de todos los elementos x para los cuales  $x+b \in A$  para todo  $b \in B$ . esto es;

$$A \ominus B = \{x \in E^n : x + b \in A, \forall b \in B\} \quad (4.30a)$$

En otras palabras, la erosión de A por B es el conjunto de todos los elementos x de  $E^n$  para los cuales B trasladado a x está contenido completamente en A.

$$A \ominus B = \{x \in E^n : Bx \subseteq A\}$$



**Figura 4.25.** Elemento estructurante 4-conexo

La imagen RGB es convertida a escala de grises eliminando el Hue y la información de saturación, reteniendo la luminiscencia. En otras palabras se convierte el color verdadero de RGB de la imagen a una imagen de intensidad en la escala de grises. Además, la imagen RGB es convertida a coordenadas NTSC, fijando los componentes Hue y de saturación a cero.

#### **IV.18.3 SEGMENTACIÓN [4.35,4.36,4.37].**

Para la segmentación de la imagen se parte del hecho de tener buen contraste entre los objetos de interés y el fondo, es por ello que se aplica una de las técnicas más sencillas: la binarización. Este aspecto no resta generalidad a la solución del problema por ser algo alcanzable aún en un ambiente industrial. La binarización consiste en partir en dos el histograma de la imagen: todos los píxeles del fondo se asocian a un solo valor de gris y los píxeles del objeto se pueden asociar también a un valor de gris o bien dejarlos como estaban. El umbral es fijado de forma manual o dependiendo de la intensidad de la luz (automática), no obstante, consume mucho tiempo de cómputo. Al inicializarse la ejecución del sistema de visión, el umbral es inicializado en un nivel de gris de 0.3, el cual se encontró aceptable en las condiciones de iluminación del laboratorio. Este valor puede ser ajustado hasta lograr un nivel de binarización adecuado.

La imagen es convertida a una imagen binaria, basada en un valor de umbralización. Esta imagen generada contiene ceros y unos. Ceros para representar el color negro y 1 para otra intensidad de color.

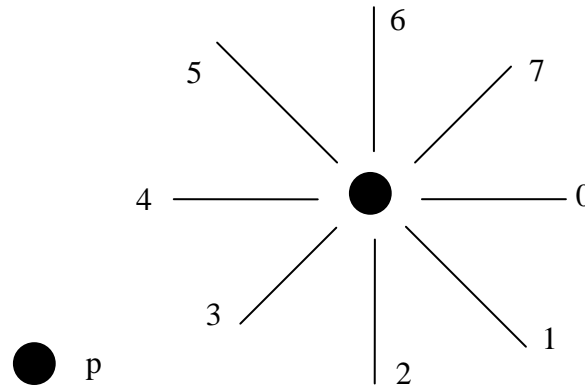
#### **IV.18.4 RECONOCIMIENTO [4.38,4.39,4.43]**

Para la descripción y posterior reconocimiento de los objetos dentro del campo de visión de la cámara, es necesario establecer un vector de atributos que permita la diferenciación de cada clase u objeto. Los componentes de este vector de atributos son:

- El factor de compactación
- Los momentos invariantes

Tales componentes fueron elegidos por la robustez que ofrecen al objetivo. Además, de la velocidad de cálculo que se alcanza con los algoritmos utilizados. Partiendo del hecho de ser el contorno una de las formas más naturales y sencillas de representar los objetos que nos rodean, se estableció un algoritmo de seguimiento de contornos para

regiones conexas, el cual a su vez proporciona información correspondiente a la región encerrada que finalmente posibilita el cálculo exacto de los componentes del vector de atributos. El algoritmo de seguimiento de contornos empleado adopta un sentido de recorrido horario de acuerdo a la forma en que se establecen los 8 vecinos del píxel analizado y que se muestra en la figura 4.26.



**Figura 4.26.** Vecino 8-conexos del punto p.

El factor de compactación es un atributo geométrico de los objetos que son muy empleados en tareas de clasificación por su invarianza a transformaciones tales como: traslación, rotación y cambios de escala. Este factor se define de la siguiente forma:

$$f_{comp} = \frac{P^2}{A} \tag{4.30b}$$

Tabla 4.8. Incrementos al perímetro	
Dirección	Valor agregado
0	Largo
1	Diagonal
2	Ancho
3	Diagonal
4	Largo
5	Diagonal
6	Ancho
7	diagonal

Donde:

*f<sub>comp</sub>* : Factor de compactación, donde P es el perímetro y A denota el área.

Para la obtención del perímetro del objeto se emplea un algoritmo de seguimiento de contornos. Este toma en cuenta la dirección en que son encontrados los píxeles figura 4.26 y tabla 4.8, para conocer los incrementos que deben sumarse a un acumulador iniciado en cero. De esta manera, al término del recorrido el acumulador contendrá el valor del perímetro.

En la tabla 4.8, el largo, ancho y diagonal representan las dimensiones del píxel, las cuales fueron encontradas de manera experimental y los valores fueron: 1.44, 0.88 y 1.04 respectivamente. El área del objeto es obtenida a través del algoritmo de



computación de los momentos invariantes, representada por el momento de orden 0 ( $m_{00}$ ). Es importante mencionar, que la imagen ya binarizada, debe pasar por un proceso de filtrado para eliminar el ruido y las impurezas. Esto se logra con un algoritmo que ignora objetos pequeños con un número de píxeles menores a P. Se eliminan grupos de píxeles conectados (objetos), produciendo otra imagen binaria, en donde la conectividad es de 8 por dos dimensiones.

Después la imagen que se obtiene, es convertida a su complemento. Esto quiere decir, que en la imagen binaria, los ceros son convertidos a 1's, y los 1's a ceros. El blanco y el negro son invertidos. En el complemento de una imagen RGB o de intensidad, cada valor de píxel es sustraído del máximo valor soportado por la clase. (o 1.0 de imágenes de doble precisión), y la diferencia es usada como el valor del píxel en la imagen de salida. En la imagen de salida las áreas negras son convertidas a blancas y las blancas en negras.

Un ejemplo:

```
X = uint8([ 255 10 75; 44 225 100]);
X2 = imcomplement(X)
X2 =
    0 245 180
   211 30 155
```

#### IV.18.5 MOMENTOS INVARIANTES [4.39,4.40,4.41,4.42,4.43]

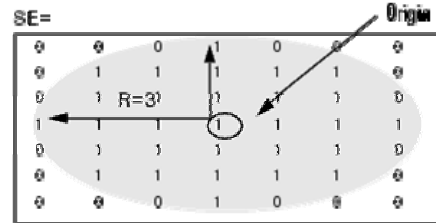
La teoría de momentos provee una alternativa usual en la representación de formas u objetos. El concepto de momentos invariantes se basa en la teoría de invariantes, la cual trata con las propiedades de ciertas clases de expresiones algebraicas que no varían a transformaciones lineales, tales como la rotación, traslación y escalamiento. Los momentos invariantes han sido utilizados para describir o caracterizar objetos en procesos de reconocimiento cuando se tiene unicidad de forma pero sin perspectiva, independientemente de su localización, medida y orientación.

Para obtener los momentos invariantes se optó por emplear un algoritmo basado en el contorno de un objeto, ya que como se mencionó anteriormente, requiere menos tiempo de cálculo. Además, de proporcionar resultados exactos. La imagen es transformada a un elemento morfológico estructurado, tipo disco.

Por ejemplo:

`SE = strel('disk',R,N)`, crea un elemento estructurado en forma de disco plano, donde R, especifica el radio. R debe ser un número no negativo. N debe tomar valores: 0, 4, 6 y 8. cuando N es mayor que cero, el elemento estructurado en forma de disco es aproximado por una secuencia de N elementos estructurados periódicos lineales. Cuando N es igual a cero, la aproximación no se utiliza, y los miembros de los elementos estructurantes consisten en todos los píxeles cuyos centros no son mayores a R desde el origen. Si N no se especifica, El valor por *default* es 4. Es importante

mencionar, que las operaciones morfológicas corren más rápido cuando los elementos estructurantes utilizan aproximaciones ( $N>0$ ) que cuando ( $N=0$ ).



**Figura 4.27** Elemento estructurante

Después la imagen se cierra convirtiendo en un objeto estructurado. Una vez que se tiene el objeto estructurado, se hace pasar un algoritmo que rellena hoyos en la imagen binaria. Un hoyo es un conjunto de píxeles del fondo que no fueron alcanzados a ser rellenados por el borde la imagen. Luego, se crean contornos alrededor de los objetos creados morfológicamente. Aquí, la imagen debe ser binaria donde los píxeles diferentes de cero pertenecen al objeto y los píxeles con valor cero pertenecen al fondo (background). Después, la imagen etiquetada es convertida a una imagen tipo RGB, con el propósito de visualizar las regiones etiquetadas. Luego, se obtienen las propiedades de interés, en este caso: el centroide de cada objeto, el área y la orientación respectiva.

#### IV.18.6 ANÁLISIS DE RASGOS DISTINTIVOS [4.35]

El análisis de rasgos distintivos es importante, en este caso, debido a lo fácil y rápido que resulta su implementación. Muchos problemas prácticos en la industria pueden resolverse con análisis de rasgos distintivos. La tarea esencial del sistema de visión es analizar la posición y orientación de un bloque rectangular como rasgos distintivos. Se consideran dos bloques dentro del campo de visión y dentro del campo de trabajo del robot interceptor.

Para determinar el centro y la orientación de una región rectangular de dimensiones conocidas con el análisis de rasgos distintivos, como se muestra en la figura 4.21. Se divide el rectángulo en seis partes formadas por dos líneas verticales, a una distancia conocida  $g$ , y una línea horizontal, como se muestra en la figura 4.22. Esta división corresponde a la proyección indexada de la imagen. Se asume que las esquinas del rectángulo están en los extremos de las sextas partes. Es decir, la esquina superior izquierda es la sexta etiqueta A, figura 4.23, la esquina inferior izquierda, es la etiqueta B y así sucesivamente. Las dimensiones  $h$  y  $w$  son conocidas. La división de partes constituyen la proyección indexada de la imagen. El área de intersección del rectángulo con cada sexta parte se determina enmascarando la proyección indexada de la imagen con la imagen binarizada del rectángulo. Un histograma del enmascaramiento de la proyección de la imagen (el rasgo distintivo) suministra el área de intersección de cada sexta parte del rectángulo. El problema es como usar las seis áreas para determinar las unidades y orientación del rectángulo.

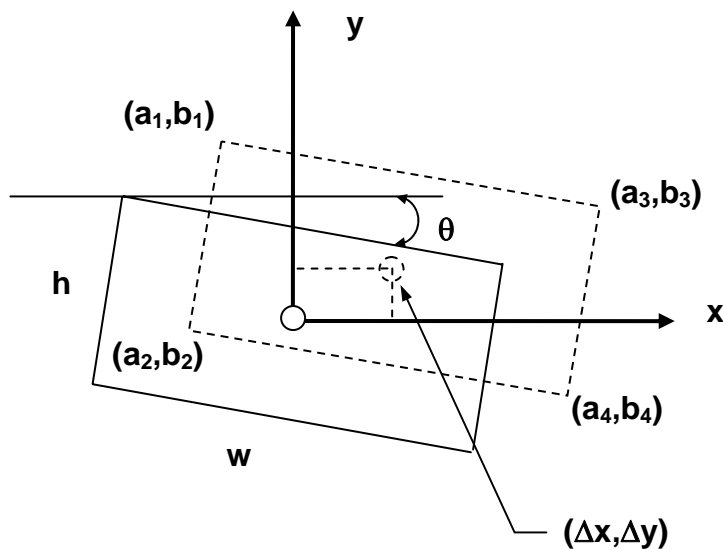
Para resolver el problema, se fija un sistema de coordenadas x-,y cuyo origen esté al centro de las seis celdas del rectángulo dividido. En la práctica puede existir un número mayor de particiones de la imagen. Primero se supone que las coordenadas  $(a_1, b_2)$  son conocidas y que el ángulo  $\theta$  también. Para conocer las coordenadas  $(\Delta x, \Delta y)$  del centro del rectángulo. Primero se determinan  $(\Delta x, \Delta y)$  en términos del segmento de la línea  $v$  desde la esquina superior izquierda del rectángulo con la línea horizontal y el segmento de la línea  $u$  desde la esquina superior izquierda del rectángulo a la línea vertical que se encuentra más a la izquierda. En la geometría que se muestra en la figura 4.22 se aprecian los resultados.

Para una rotación en el sentido de las manecillas del reloj, un punto  $(x, y)$  es rotado al punto  $(x_{rot}, y_{rot})$ , donde:

$$\begin{pmatrix} x_{rot} \\ y_{rot} \end{pmatrix} = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

Para la rotación y el cambio de  $(\Delta x, \Delta y)$ , el punto  $(x, y)$  se transforma al punto  $(x_{new}, y_{new})$ , donde:

$$\begin{pmatrix} x_{new} \\ y_{new} \end{pmatrix} = \begin{pmatrix} x_{rot} \\ y_{rot} \end{pmatrix} + \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix}$$



$$\begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} = \begin{pmatrix} a1 + \frac{w}{2} \cos \theta - \frac{h}{2} \sin \theta \\ b1 + \frac{w}{2} \sin \theta - \frac{h}{2} \cos \theta \end{pmatrix}$$

**Figura 4.28.** Geometría para determinar la traslación del centro del rectángulo en términos de la posición de una de esquinas el centro de

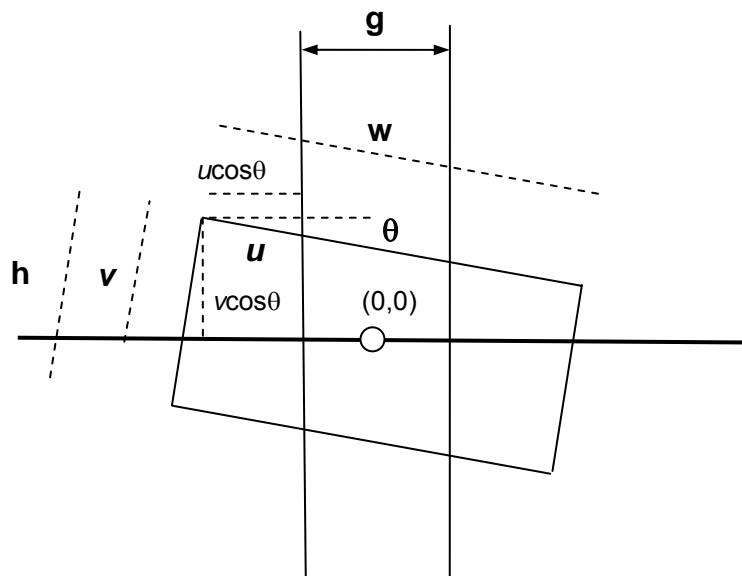
Para el caso de la esquina  $(a_1, b_1)$ , se tiene de la figura 4.28.

$$\begin{aligned} \begin{pmatrix} a_1 \\ b_1 \end{pmatrix} &= \begin{pmatrix} \cos\theta & \sin\theta \\ -\operatorname{sen}\theta & \cos\theta \end{pmatrix} \begin{pmatrix} -\frac{w}{2} \\ \frac{h}{2} \end{pmatrix} + \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \\ &= \begin{pmatrix} -\frac{w}{2}\cos\theta + \frac{h}{2}\sin\theta \\ \sin\theta + \frac{h}{2}\cos\theta \end{pmatrix} + \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \end{aligned}$$

De donde se obtiene: 
$$\begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} = \begin{pmatrix} a_1 + \frac{w}{2}\cos\theta - \frac{h}{2}\sin\theta \\ b_1 + \frac{w}{2}\sin\theta - \frac{h}{2}\cos\theta \end{pmatrix}$$

Sustituyendo  $-g/2 - u\cos\theta$  por  $a_1$  y  $v\cos\theta$  por  $b_1$ , se obtiene:

$$\begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} = \begin{pmatrix} -\frac{g}{2} - u\cos\theta + \frac{w}{2}\cos\theta - \frac{h}{2}\sin\theta \\ v\cos\theta - b_1 + \frac{w}{2}\sin\theta - \frac{h}{2}\cos\theta \end{pmatrix} \quad (4.31)$$



$$\begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} = \begin{pmatrix} -\frac{g}{2} - u\cos\theta + \frac{w}{2}\cos\theta - \frac{h}{2}\sin\theta \\ v\cos\theta - \frac{w}{2}\sin\theta - \frac{h}{2}\cos\theta \end{pmatrix}$$

**Figura 4.29.** Geometría para determinar la traslación del centro del rectángulo en términos de las longitudes  $u$  y  $v$ .

Después se determinan las longitudes  $u$  y  $v$  en términos de las áreas calculadas A, B, C, D, E y F como se muestran en la figura 4.29. De la geometría inmediatamente se derivan las siguientes ecuaciones:

$$A + B = uh + \frac{1}{2}h^2 \tan \theta$$

$$C + D = \left(w - \frac{g}{\cos \theta} - u\right)h - \frac{1}{2}h^2 \tan \theta$$

Se utilizan ambas expresiones para mantener la simetría y la estabilidad numérica para determinar una expresión para  $u$ . Restando la primera ecuación de la segunda, se obtiene:

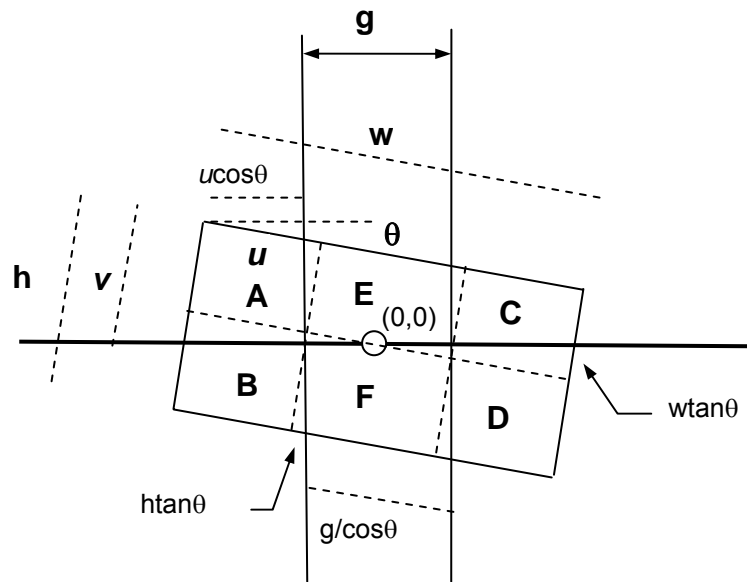
$$(C + D) - (A + B) = \left(w - \frac{g}{\cos \theta}\right)h - 2uh - h^2 \tan \theta$$

Despejando el término  $2uh$

$$2uh = -(C + D) + (A + B) + \left(w - \frac{g}{\cos \theta}\right)h - h^2 \tan \theta$$

Entonces:

$$u = \frac{(A + B) - (C + D)}{2h} + \frac{1}{2}\left(w - \frac{g}{\cos \theta}\right)h - \frac{1}{2}h \tan \theta \quad (4.32)$$



**Figura 4.30.** Geometría para determinar las longitudes  $u$  y  $v$  en términos de las áreas calculadas A, B, C, D, E y F.

También de la figura 4.30, se tiene:

$$A + E + C = vw - \frac{1}{2}w^2 \tan \theta$$

$$B + F + D = (h - v)w + \frac{1}{2}w^2 \tan \theta$$

De nuevo se utilizan ambas ecuaciones para mantener la simetría y la estabilidad numérica para determinar una expresión para  $v$ . Restando la primera de la segunda, se obtiene:

$$\begin{aligned}(A + E + C) - (B + F + D) &= vw - \frac{1}{2}w^2 \tan \theta - hw + vw - \frac{1}{2} \tan \theta \\ &= -2vw - hw - w^2 \tan \theta\end{aligned}$$

Resolviendo para  $v$ , se obtiene:

$$\begin{aligned}2vw &= (A + E + C) - (B + F + D) + hw + w^2 \tan \theta \\ v &= \frac{(A + E + C) - (B + F + D) + \frac{h}{2} + \frac{w}{2} \tan \theta}{2w}\end{aligned}\quad (4.33)$$

Sustituyendo las ecuaciones (4.32) y (4.33) en términos de las áreas calculadas  $A$ ,  $B$ ,  $C$ ,  $D$ ,  $E$  y  $F$  en la ecuación (4.31), se obtiene:

$$\begin{aligned}\Delta x &= \frac{g}{2} - u \cos \theta + \frac{w}{2} \cos \theta - \frac{h}{2} \sin \theta \\ \Delta x &= \frac{g}{2} - \left[ \frac{(A + B) - (C + D)}{2h} + \frac{1}{2} \left( w + \frac{g}{\cos \theta} \right) - \frac{1}{2} h \tan \theta \right] \cos \theta + \frac{w}{2} \cos \theta - \frac{h}{2} \sin \theta \\ \Delta x &= \frac{(C + D) - (A + B)}{2h} \cos \theta\end{aligned}\quad (4.34)$$

Y para  $\Delta y$ :

$$\begin{aligned}\Delta y &= v \cos \theta - \frac{w}{2} \sin \theta - \frac{h}{2} \cos \theta \\ &= \left[ \frac{(A + E + C) - (B + F + D)}{2w} + \frac{h}{2} + \frac{w}{2} \tan \theta \right] \cos \theta - \frac{w}{2} \sin \theta - \frac{h}{2} \cos \theta \\ &= \frac{(A + E + C) - (B + F + D)}{2w} \cos \theta + \frac{h}{2} \cos \theta + \frac{w}{2} \sin \theta - \frac{w}{2} \sin \theta - \frac{h}{2} \cos \theta \\ \Delta y &= \frac{(A + E + C) - (B + F + D)}{2w} \cos \theta\end{aligned}\quad (4.35)$$

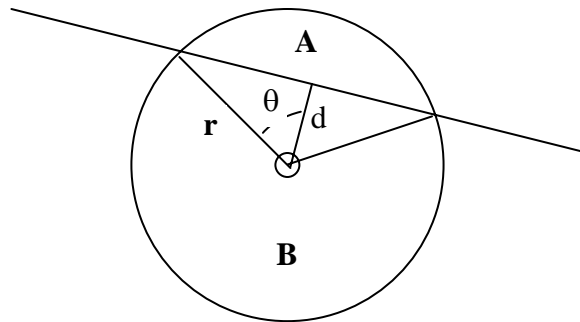
Finalmente, es fácil determinar el ángulo de rotación  $\theta$  en términos de las áreas  $E$  y  $F$ , el cual constituye un paralelogramo. De la geometría se tienen que:

$$E + F = \frac{hg}{\cos \theta} \quad \text{de aquí,} \quad \cos \theta = \frac{hg}{E + F}$$

## IV.18.7 DETERMINACIÓN DEL CENTRO DE UN CÍRCULO [4.35]

Para establecer la posición de una región circular con análisis de características o rasgos distintivos, primero se divide el círculo en cuatro cuadrantes formados por dos líneas ortogonales garantizando que se encuentren dentro del círculo, luego, se calcula el área de cada cuadrante con el histograma de la proyección indexada de la imagen enmascarada que consiste de los cuatro cuadrantes de un círculo. Para entender como se convierte el área en información de posición, se considera la situación resultante cuando una cuerda parte en dos regiones a un círculo A y B. Se supone que la cuerda se encuentra a una distancia  $d$  del centro del círculo y que el radio del círculo es  $r$ .

Sea  $\theta$  el ángulo entre la línea perpendicular que corta la cuerda y el segmento de línea que parte del origen del círculo, ver figura 4.31.



**Figura 4.31.** Geometría del círculo, su centro y una cuerda.

Entonces, el área total de los dos triángulos es  $d\sqrt{r^2 - d^2}$  y el área del sector con un ángulo central  $2\theta$  y radio  $r$  es  $r^2\theta$ . El ángulo es dado por:  $\theta = \cos^{-1} \frac{d}{r}$ . Por lo tanto, el área del segmento determinado por la cuerda y la circunferencia es:

$$A = r^2 \cos^{-1} \frac{d}{r} - d\sqrt{r^2 - d^2}$$

$$A = r^2 \left[ \cos^{-1} \frac{d}{r} - \frac{d}{r} \sqrt{1 - \left(\frac{d}{r}\right)^2} \right]$$

Donde  $\theta = \cos^{-1} \frac{d}{r}$ , se puede reescribir el segmento de área.

$$A = \frac{r^2}{2} [2\theta - \sin 2\theta] \quad (4.36)$$

Y el área total del círculo es:

$$A + B = \pi r^2$$

De aquí, el radio del círculo se puede determinar por;

$$r = \sqrt{\frac{A+B}{\pi}} \quad (4.37)$$

Sustituyendo la ecuación (4.37) en la (4.36), se obtiene:

$$\frac{2\pi A}{A+B} = 2\theta - \sin 2\theta \quad (4.38)$$

Esta ecuación trascendental no tiene una solución cerrada para el ángulo  $\theta$ . Se puede aproximar por técnicas numéricas. Una vez calculado  $\theta$ , el *offset*  $d$  se determina por:

$$d = \sqrt{\frac{A+B}{\pi}} \cos \theta \quad (4.39)$$

Ahora, ya se puede considerar el problema original de determinar el centro del círculo en cualquier posición. Midiendo las áreas A, B, C y D. ver figura 4.32.

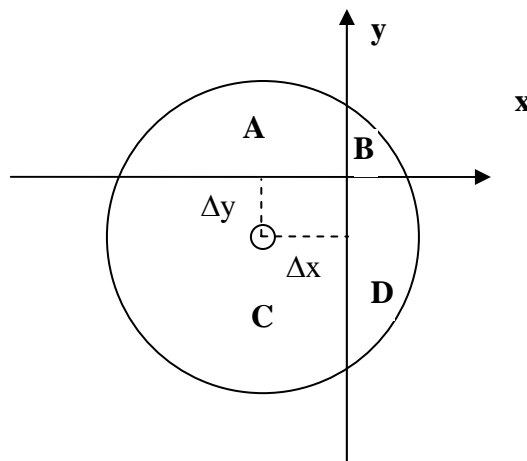
Sí  $A+B > C+D$ , entonces la coordenada  $y$  del centro del círculo es positiva; de otro modo, es negativa. Sí  $B+D > A+C$ , entonces la coordenada  $x$  del centro del círculo es positiva, de la otra forma, es negativa.

La magnitud de la coordenada  $y$  está dada por:

$$|\Delta y| = \sqrt{\frac{A+B+C+D}{\pi}} \cos \theta_y \quad (4.40)$$

donde  $\theta_y$ , satisface:

$$\frac{2\pi(A+B)}{A+B+C+D} = 2\theta_y - \sin 2\theta_y$$



**Figura 4.32.** Círculo proyectado sobre los cuatro cuadrantes de la proyección indexada de la imagen.

La magnitud de la coordenada  $x$  está dada por:



$$|\Delta x| = \sqrt{\frac{A+B+C+D}{\pi} \cos \theta_x} \quad (4.41)$$

donde  $\theta_x$  satisface:

$$\frac{2\pi(B+D)}{A+B+C+D} = 2\theta_x - \sin 2\theta_x$$

#### IV.19 RESULTADOS EXPERIMENTALES DE LA IMPLEMENTACIÓN DEL SISTEMA DE VISIÓN.

En este tema se implementó el sistema de visión con la finalidad de obtener algunos parámetros cinemáticos de las piezas en movimiento. Los rasgos distintivos a calcular son el centroide (localizado en el centro de área), La orientación con respecto al eje horizontal y el área en pixeles. En este caso serán dos objetos: un rectángulo y un círculo, de dimensiones conocidas. Es importante mencionar que el algoritmo desarrollado para este fin, también puede encontrar los rasgos distintivos para más objetos y para formas irregulares con el fin de no perder generalidad.

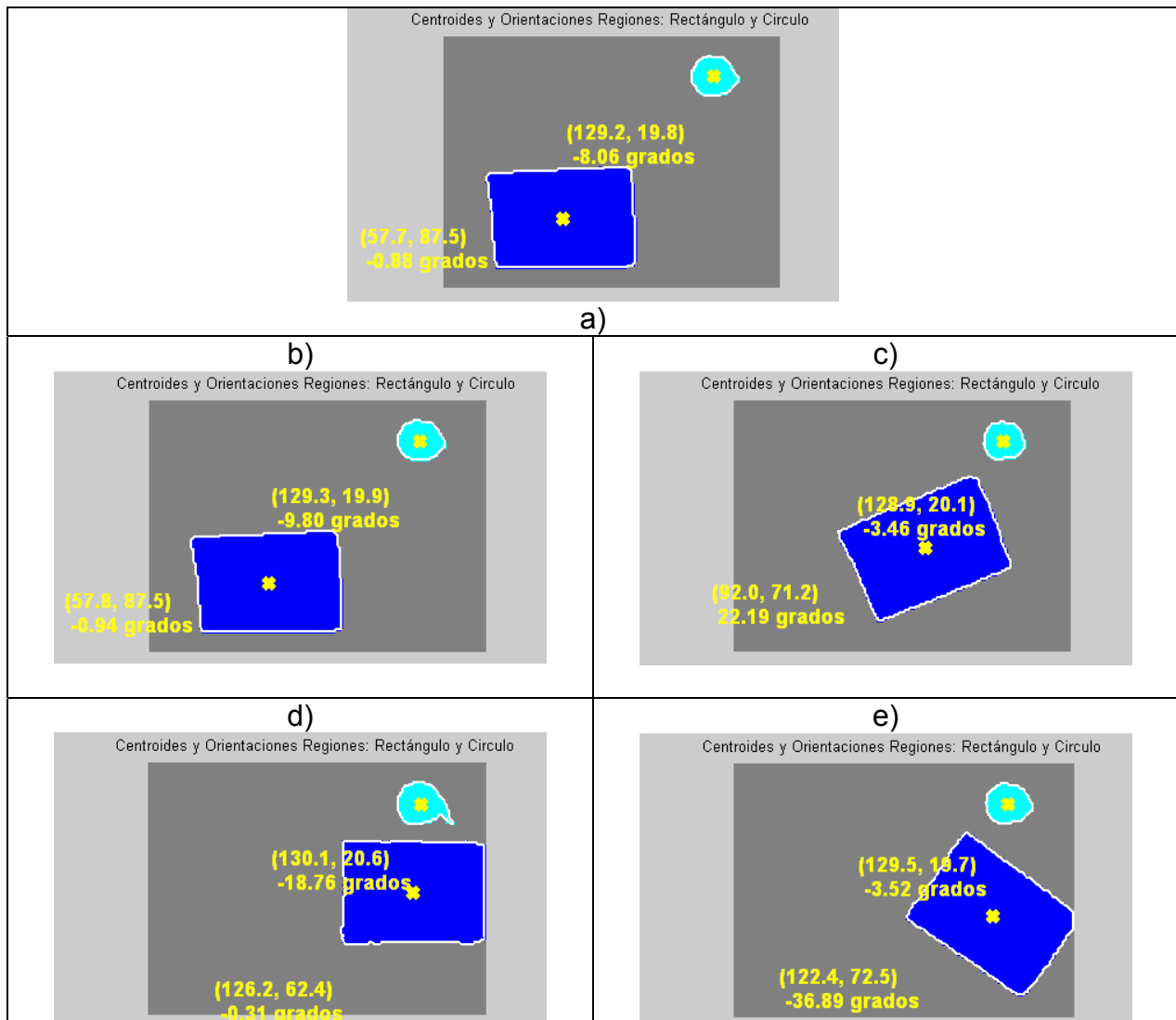
El experimento consistió en determinar el movimiento de un rectángulo y un círculo estático.



**Figura 4.33.** Imagen original y binarizada con un umbral de 0.25 en una escala de 0 a 1

En condiciones de luz desfavorables [4.36], En la figura 4.33a, se muestra la imagen original con un rectángulo color negro de 100x70 mm y un objeto circular color azul marino con un radio de 30 mm. El objeto circular pertenece al estuche de unos lentes de contacto y se realizó la binarización de tal forma que el color blanco se confundiera

con el color blanco del fondo. El umbral fue de 2.5, en una escala del 0 al 1. Se puede observar en la figura 4.33b, que existen algunas impurezas (ruido), y que será necesario limpiarlas aplicando un filtro. Cabe mencionar, que las sombras provocadas por la luz incidente de un foco incandescente y la combinación de luz ambiental, constituye un factor de error en la binarización. La distancia de la cámara se encuentra en un plano paralelo, al plano de la mesa donde se encuentran los objetos de 200 mm.



**Figura 4.34.** Seguimiento de la posición de los centroides y ángulos de orientación de los objetos.

---

## IV.20 SUMARIO

Se ha presentado el prototipo de robot interceptor y el prototipo del disparador para dos partículas, estos dos dispositivos permitirán realizar eventos experimentales de intercepción. Por el momento, solo se plantea la calibración del sistema de visión. El disparador permite conocer el ángulo de disparo y la velocidad de salida de las partículas libres TPL1 y TPL2. Se ha presentado también el esquema de control de movimiento de los eslabones del hombro y del codo del robot interceptor, que son del tipo (PID). Es importante mencionar que el efector final solo tiene implementado un control tipo P (Proporcional) y solo orienta la red para interceptar los objetos. Se explica en breve los códigos de programación en el entorno de programación para recibir y enviar señales analógicas y digitales. El tipo de movimiento que se enviarán a los motores serán perfiles de velocidad tipo trapezoidal, con aceleración constante.

La propiedad de linealidad del sistema de visión establece que a pesar de la naturaleza no lineal en términos de los parámetros extrínsecos e intrínsecos (incluyendo distorsión radial), el modelo de visión puede ser expresado como un modelo de regresión lineal en términos de un conjunto adecuado de parámetros compuesto por la combinación de parámetros extrínsecos e intrínsecos. Esta propiedad de linealidad es válida cuando la cámara se encuentra rotada alrededor de uno de los ejes principales del sistema de referencia del robot manipulador, en este caso el robot interceptor.

A partir de la propiedad de linealidad del sistema de visión de manera natural se desprende un algoritmo de calibración que incluye un conjunto de expresiones en forma cerrada para calcular los parámetros extrínsecos en forma explícita y de manera indirecta los parámetros intrínsecos.

Una ventaja que presentan los controladores en la configuración cámara fija es que el único parámetro del sistema de visión que se necesita conocer es el ángulo  $\theta$  que denota la orientación del sistema de referencia de la cámara con respecto al sistema de referencia del robot.

## IV.21 REFERENCIAS

- [4.1] Sciavicco L. y B. Siciliano, "Modeling and Control of Robot Manipulators", Mc Graw Hill. Primera edición. New York, 1996.
- [4.2] An C.H., C. G. Atkeson y J. M. Hollerbach, "Model-Based Control of a Robot Manipulator", The MIT Press, 1988.
- [4.3] Jager B.D. y J. Banens, "Experimental evaluations of robot controllers", Proceedings of the 33<sup>rd</sup>. Conference on decision and control. Lake Buena Vista, Fl. U.S.A. pp: 363-368, December 1994.
- [4.4] Corke P.I. and B. Armstrong, "A meta-study of PUMA 560 dynamics: A critical appraisal of literature data", *Robotica*, Vol. 13, pp:253-258, 1995.
- [4.5] Slotine J. J. y W. Li, "Adaptive Manipulator Control: A case study", *IEEE Transactions on Automatic Control*, 33(11), pp: 995-1003, 1988.
- [4.6] Asada H. y K. Youcef-Toumi, "Direct-Drive Robots: Theory and Practice", The MIT Press, 1987.
- [4.7] Germán V. M., "Cinemática y dinámica de la partícula", Instituto Politécnico Nacional, Dirección de Publicaciones, 2002. ISBN 970-18-8146-X.
- [4.8] Michael D. Naish, Elizabeth A. Croft, Beno Benhabib, "Coordinate dispatching of proximity sensors for the surveillance of manoeuvring targets", *Robotics and Computer Integrated Manufacturing*, Vol. 19, pags:283-299, 2003.
- [4.9] Anne Wright, Randy Sargent, *et al*, *Cognachrome Vision System User's Guide*, Manual Edition 2.1, Newton Research Labs, 1998.
- [4.10] Corke P.I., "Visual control of robot manipulators – a review". *Visual servoing: Real-time control of robot manipulators based on visual sensory feedback*, Edited by Koichi Hashimoto. World Scientific Series in Robotics and Automated Systems, Vol. 7, pags: 1-31, 1993.
- [4.11] Papanikopoulos P. y P. K. Hhosla and T. Kanade, "Visual tracking of moving target by a camera mounted on a robot: A combination of control and vision", *IEEE Transaction on Robotics and Automation*, Vol.9,No.1, pags: 14-35, 1993.
- [4.12] Wijesoma W. S., D. F.H. Wolfe and R. J. Richards, "Eye to hand coordination for vision-guided robot control applications", *The international J. Robot Res.*, Vol.12, No.1, pags: 65-78, 1993.
- [4.13] Carrelli R. O. Nassisi and B. Kuchen, "Adaptive robot control with visual feedback", *Proceedings of the American Control Conference*. Baltimore. Maryland. June, pags: 1757-1760, 1994.
- [4.14] Hashimoto K. y H. Kimura, "Dynamical visual servoing with nonlinear model-based control", *IFAC World Congress*. Sidney. Australia. Vol. 9, pags: 405-408, 1993.
- [4.15] Hashimoto K. y H. Kimura, "Visual servoing with nonlinear observer", *Proceedings of the IEEE International Conference on Robotics and Automation*, pags: 484-489, 1995.
- [4.16] Kelly R., "Robust asymptotically stable visual servoing of planar robots", *IEEE Transactions on Robotics and Automation*, Vol. 12, No. 5, pags: 759-766, October, 1996.
- [4.17] Kelly R., P. Shirkey y M. W. Spong, "Fixed camera visual servo control for planar robots", *Proceedings of the IEEE International Conference Robotics and Automation*, Vol. 3, pags: 2463-2469. April 22-28, 1996.
- [4.18] Espiau B., F. Chaumette and P. Rives, "A new approach to visual servoing in robotics", *IEEE Transactions on Robotics and Automation*, Vol. 8, No. 3, pags: 313-326, June 1992.
- [4.19] Preising B. y T. C. Hsia, "Robot performance measurement and calibration using a 3D computer vision system", *Robotics*, pags: 327-337, 1995.
- [4.20] Lenz K. R. y R. Y. Tsai, "Techniques for calibration of the scale factor and image center for high accuracy 3D machine vision metrology", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 10, No. 5, pags: 713-720, September 1988.
- [4.21] Spong M. y M. Vidyasagar, "Robot Dynamics and Control", John Wiley and Sons. Primera Edición, New York, 1989.

- [4.22] Horn P. K., "Robot Vision", The MIT Press, Cambridge. MA, 1986.
- [4.23] Sciavicco L. y Siciliano, "Modeling and Control of Robot Manipulators", Mc Graw Hill, Primera Edición, New York, 1996.
- [4.24] Klein M. V. y T. E. Furtak, "Optics", John Wiley & Sons, 1986
- [4.25] Weng J., P. Cohen y M. Herniou, "Camera calibration with distortion models and accuracy evaluation", IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 14, No. 10, pags: 965-980, October 1992.
- [4.26] Nomura Y., M. Sagara, H. Naruse and A. Ide, "Simple calibration algorithm for high distortion lens camera", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, No. 11, pags: 1095-1099, November 1992.
- [4.27] Feddema J. T., C. S. G. Lee and O. R. Mitchell, "Weighted selection of image features for resolved rate visual feedback control", IEEE Transactions on Robotics and Automation, Vol. 7, No. 1, pags: 31-47, Febrero 1991.
- [4.28] José Fernando Reyes Cortés, "Control de un robot de transmisión directa de dos grados de libertad", tesis doctoral, Centro de Investigación científica y de educación superior de Ensenada, Baja California, México, 1997.
- [4.29] Tekalp A. M., "Digital video processing", Primera edición, Prentice Hall, 1995.
- [4.30] Hung Y. P., "A simple real-time method for calibrating a camera mounted on a robot for three dimensional machine vision", Optics, illumination and Image Sensing for Machine Vision III, SPIE Vol. 1005, pags: 12-19, 1988.
- [4.31] Tsai R. Y., "A versatil camera calibration technique for high-acurracy 3D machine vision metrology using off-the-shelf TV camera and lens", IEEE Journal of Robotics and Automation, Vol. RA-3, No. 4, pags: 323-344, Agosto 1987.
- [4.32] Zhuang H. y W. C. Wu, "Camera Calibration with a Near Parallel (III-Conditioned) Calibration Board Configuration", IEEE Transactions on Robotics and Automation, Vol. 12, No. 6, pags: 918-921, Diciembre 1996.
- [4.33] Reyes F. y R. Kelly, "On vision systems identification", Reporte Técnico, CICESE, Junio 1997.
- [4.34] Soille, P., Morphological Image Analysis: Principles and Applications, Springer-Verlag, 1999, pp. 173-174.
- [4.35] Robert M. Haralick, Linda G. Shapiro, "Computer and Robot Vision", Addison-wesley Publishing Company, 1992.
- [4.36] Arturo de la escalera, "Visión por computador fundamentos y métodos", Prentice-Hall, 2001
- [4.37] Gonzalo Pajares, Jesús M. de la Cruz, "Visión por Computador Imágenes digitales y aplicaciones", Alfaomega Ra-Ma, 2002
- [4.38] J. R. Parker, "Practical Computer Vision using C", John Wiley & Sons, inc., 1994.
- [4.39] J. Lira Chávez, "Introducción al tratamiento de imágenes", Ciencia de la Computación, 2002
- [4.40] D. A. Lyon, "Image Processing in Java", Prentice-Hall Inc, 1999
- [4.41] Harley R. Myler, Arthur R. Weeks, "The Pocket Handbook of image Processing Algorithms in C", Prentice-Hall PTR, 1993.
- [4.42] Van den Boomgard, Rein, y Richard van Balen, "Methods for Fast Morphological Image Transforms Using Bitmapped Images," Computer Vision, Graphics, and Image Processing: Graphical Models and Image Processing, Vol. 54, No. 3, May 1992, pp. 252-254.
- [4.43] Adams, Rolf, "Radial Decomposition of Discs and Spheres," Computer Vision, Graphics, and Image Processing: Graphical Models and Image Processing, Vol. 55, No. 5, September 1993, pp. 325-332.
- [4.44] Jones, Ronald, y Pierre Soille, "Periodic lines: Definition, cascades, and application to granulometrie," Pattern Recognition Letters, Vol. 17, 1996, pp. 1057-1063.

---

## CAPÍTULO V

# APLICACIÓN DE LOS ALGORITMOS DE RECUPERACIÓN DE TIEMPOS AL MODELO CINEMÁTICO DEL ROBOT INDUSTRIAL MITSUBISHI®

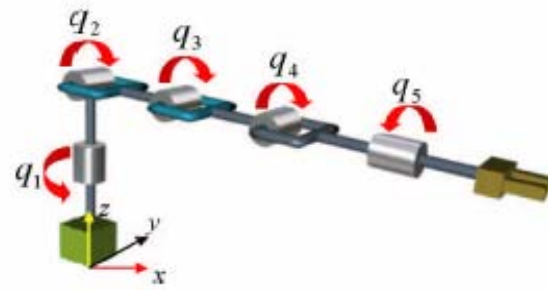
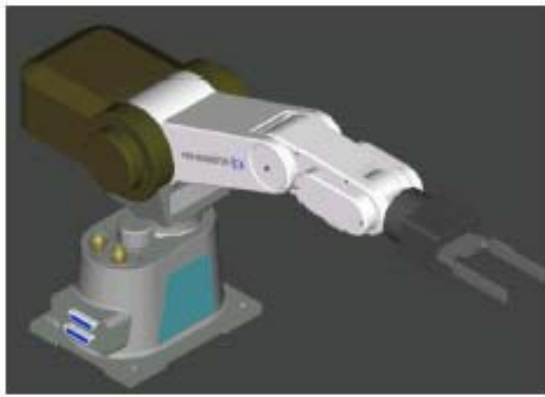
### V.1 GENERALIDADES

Este capítulo está destinado a presentar el desarrollo del modelo cinemático del robot Mitsubishi® utilizando el álgebra de Cuaterniones, con el fin de acoplar los algoritmos de recuperación de tiempos y presentar simulaciones con dicho robot. Lo anterior, permitirá llevar a cabo la parte experimental que se presenta en el capítulo VI, que consiste en presentar dos eventos para comprobar y validar los algoritmos de recuperación de tiempos desarrollados en el capítulo II: El primer experimento, consiste en interceptar dos esferas en caída libre controlada con un robot interceptor de dos grados de libertad presentado en el capítulo IV.

En el segundo experimento se implementan los algoritmos en la cinemática inversa del robot industrial Mitsubishi que alimenta los módulos de CNC (centros de maquinado) dentro de un Sistema de Manufactura Flexible (FMS), éste se encuentra formando parte de un Sistema de Manufactura Integrado por Computadora (CIM). Por lo anterior, en este tema se presentan las ecuaciones cinemáticas del robot industrial Mitsubishi® de 5 grados de libertad y algunas simulaciones en donde la intercepción múltiple se logra sin impacto (Catching).

### V.2 PARÁMETROS FÍSICOS DEL ROBOT MITSUBISHI® RV1

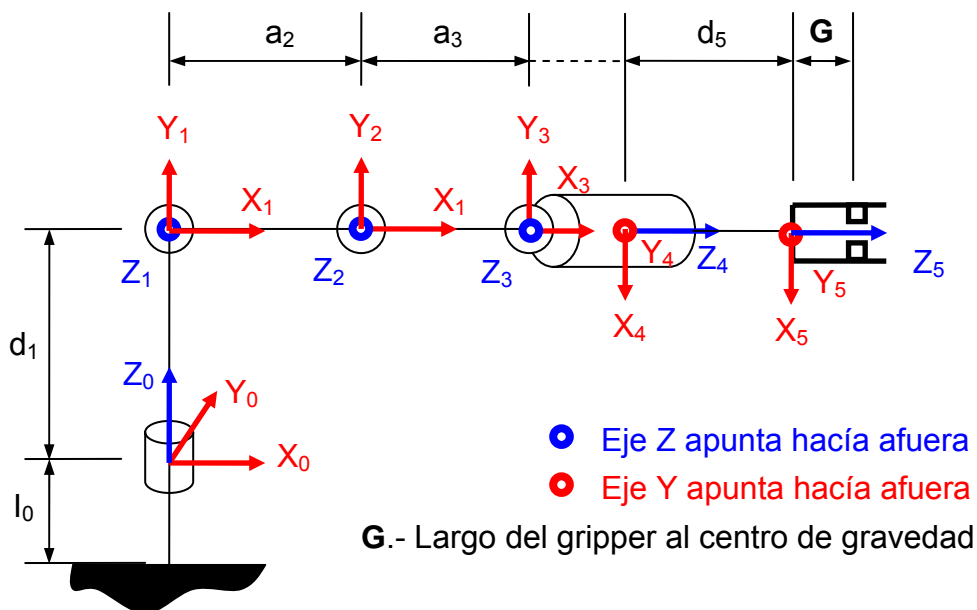
Para obtener el modelo cinemático del robot Mitsubishi es necesario contar con los parámetros físicos del robot, como son: los momentos de inercia, las longitudes, etc. El robot industrial Mitsubishi® es uno de los robots más comunes dentro de la industria y el área académica, es por ello, que se seleccionó para realizar la parte experimental. En la figura 5.1 se muestra su configuración y su área de trabajo en coordenadas articulares.



Coordenada generalizada	Mínimo (°)	Máximo (°)
q1	150	-150
q2	110	-130
q3	0	-110
q4	90	-90
q5	180	-180

**Figura 5.1.** Representación esquemática del robot Mitsubishi® y sus ángulos de desplazamiento máximos.

En la figura 5.2 se ilustra la orientación de los sistemas de coordenadas y los parámetros definidos bajo la convención de Denavit-Hartenberg.



**Figura 5.2** Representación de los sistemas de coordenadas en el robot Mitsubishi®.

**Tabla 5.1** Parámetros de Denavit-Hatemberg

Articulación		$q_i$	$\alpha_i$	$a_i$	$d_i$
	1	$q_1$	90	0	$d_1$
	2	$q_2$	0	$a_2$	0
	3	$q_3$	0	$a_3$	0
	4	$q_4$	90	0	0
	5	$q_5$	0	0	$d_5$

Parámetros físicos del robot Mitsubishi RV1

**Tabla 5.2** Parámetros físicos del robot

Longitud	Notación	Valor	Unidades
Eslabón 1	$d_1$	0.300	m
Eslabón 2	$a_2$	0.250	m
Eslabón 3	$a_3$	0.160	m
Eslabón 5	$d_5$	0.072	m

**Tabla 5.3** Parámetros dinámicos del robot

Descripción		Notación	Valor	Unidad
Masas	Eslabón 1	$m_1$	9	kg
	Eslabón 2	$m_2$	5	kg
	Eslabón 3	$m_3$	3	kg
	Eslabón 4	$m_4$	0	kg
	Eslabón 5	$m_5$	2	kg
Centro de masa	Eslabón 1	$l_{c_1}$	0.150	m
	Eslabón 2	$l_{c_2}$	0.125	m
	Eslabón 3	$l_{c_3}$	0.080	m
	Eslabón 4	$l_{c_4}$	0	m
	Eslabón 5	$l_{c_5}$	0.035	m
Momento de Inercia	Eslabón 1 en x	$I_{x_1}$	0.020	kg m <sup>2</sup>
	Eslabón 1 en y	$I_{y_1}$	0.012	kg m <sup>2</sup>
	Eslabón 1 en z	$I_{z_1}$	0.010	kg m <sup>2</sup>
	Eslabón 2 en x	$I_{x_2}$	0.010	kg m <sup>2</sup>
	Eslabón 2 en y	$I_{y_2}$	0.012	kg m <sup>2</sup>
	Eslabón 2 en z	$I_{z_2}$	0.010	kg m <sup>2</sup>
	Eslabón 3 en x	$I_{x_3}$	0.080	kg m <sup>2</sup>
	Eslabón 3 en y	$I_{y_3}$	0.018	kg m <sup>2</sup>
	Eslabón 3 en z	$I_{z_3}$	0.010	kg m <sup>2</sup>
	Eslabón 5 en x	$I_{x_5}$	0.080	kg m <sup>2</sup>
	Eslabón 5 en y	$I_{y_5}$	0.090	kg m <sup>2</sup>
	Eslabón 5 en z	$I_{z_5}$	0.090	kg m <sup>2</sup>
	Aceleración Gravedad		$g$	9.81

A continuación se presenta las matrices homogéneas de los eslabones del robot industrial Mitsubishi® 5 gdl. Esto es, el modelo cinemático de posición en términos de la notación de matrices de Denavit-Hartenberg.



$A[0,1] = \begin{pmatrix} \cos \theta_1 & 0 & \text{sen} \theta_1 & 0 \\ \text{sen} \theta_1 & 0 & -\cos \theta_1 & 0 \\ & 1 & & d_1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (5.1)$	$A[1,2] = \begin{pmatrix} \cos \theta_2 & -\text{sen} \theta_2 & 0 & \cos \theta_2 a_2 \\ \text{sen} \theta_2 & \cos \theta_2 & 0 & \text{sen} \theta_2 a_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (5.2)$
$A[2,3] = \begin{pmatrix} \cos \theta_3 & -\text{sen} \theta_3 & 0 & \cos \theta_3 a_3 \\ \text{sen} \theta_3 & \cos \theta_3 & 0 & \text{sen} \theta_3 a_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (5.3)$	$A[3,4] = \begin{pmatrix} \cos \theta_4 & 0 & \text{sen} \theta_4 & 0 \\ \text{sen} \theta_4 & 0 & -\cos \theta_4 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (5.4)$
$A[4,5] = \begin{pmatrix} \cos \theta_5 & -\text{sen} \theta_5 & 0 & 0 \\ \text{sen} \theta_5 & \cos \theta_5 & 0 & 0 \\ 0 & 0 & 1 & d_5 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (5.5)$	<p>La matriz compuesta es:</p> $T_5^0 = \begin{bmatrix} T_{11} & T_{12} & T_{13} & T_{14} \\ T_{21} & T_{22} & T_{23} & T_{24} \\ T_{31} & T_{32} & T_{33} & T_{34} \\ T_{41} & T_{42} & T_{43} & T_{44} \end{bmatrix} \quad (5.6)$

Donde:

$$\begin{aligned}
 T_{11} &= \cos \theta_1 \cos(\theta_2 + \theta_3 + \theta_4) \cos \theta_5 + \text{sen} \theta_1 \text{sen} \theta_5 \\
 T_{12} &= \cos \theta_5 \text{sen} \theta_1 - \cos \theta_1 + \cos(\theta_2 + \theta_3 + \theta_4) \text{sen} \theta_5 \\
 T_{13} &= \cos \theta_1 \text{sen}(\theta_2 + \theta_3 + \theta_4) \\
 T_{14} &= \cos \theta_1 (\cos \theta_2 a_2 + \cos(\theta_2 + \theta_3) a_3 + \text{sen}(\theta_2 + \theta_3 + \theta_4) d_5) \\
 T_{21} &= \cos(\theta_2 + \theta_3 + \theta_4) \cos \theta_5 \text{sen} \theta_1 - \cos \theta_1 \text{sen} \theta_5 \\
 T_{22} &= -\cos \theta_1 \cos \theta_5 - \cos(\theta_2 + \theta_3 + \theta_4) \text{sen} \theta_1 \text{sen} \theta_5 \\
 T_{23} &= \text{sen} \theta_1 \text{sen}(\theta_2 + \theta_3 + \theta_4) \\
 T_{24} &= \text{sen} \theta_1 (\cos \theta_2 a_2 + \cos(\theta_2 + \theta_3) a_3 + \text{sen}(\theta_2 + \theta_3 + \theta_4) d_5) \\
 T_{31} &= \cos \theta_5 \text{sen}(\theta_2 + \theta_3 + \theta_4) \\
 T_{32} &= -\text{sen}(\theta_2 + \theta_3 + \theta_4) \text{sen} \theta_5 \\
 T_{33} &= -\cos(\theta_2 + \theta_3 + \theta_4) \\
 T_{34} &= \text{sen} \theta_2 a_2 + \text{sen}(\theta_2 + \theta_3) a_3 + d_1 - \cos(\theta_2 + \theta_3 + \theta_4) d_5 \\
 T_{41} &= 0 \\
 T_{42} &= 0 \\
 T_{43} &= 0 \\
 T_{44} &= 1
 \end{aligned} \quad (5.7)$$

### V.3 MODELACIÓN CINAMÁTICA UTILIZANDO QUATERNIONES.

En este tema se presenta el desarrollo de la modelación cinemática del robot Mitsubishi® con álgebra de Cuaterniones: De la metodología presentada en el capítulo II.11 que se refiere al marco teórico. Ahora, se desarrolla en detalle para encontrar el modelo cinemático del robot utilizando la secuencia más sencilla.

Para este efecto se determinará primeramente la modelación de la configuración no deformada, que consiste en determinar un sistema de ecuaciones para calcular la posición del segundo y tercer eslabón del robot, cuando el primer eslabón se encuentra en un posición inicial de referencia (Home), conociendo la posición del extremo del tercer eslabón, que se considera para este caso como la posición del elemento terminal del manipulador.

Secuencia de rotaciones:

- 1.- Se mantiene fijo el cuarto eslabón (pinza).
- Rotación 2. Se aplica una rotación al tercer eslabón (figura 5.3)
- Rotación 3. Se aplica una rotación al segundo eslabón (figura 5.4)
4. Se mantiene fijo el primer eslabón.

De lo anterior, se define la configuración no deformada del manipulador que consiste en representar el vector de posición del elemento terminal para posiciones arbitrarias del segundo y tercer eslabón en un plano dado. Se utilizará la base canónica  $\mathfrak{R}^3$ , esto es:

$$\mathbf{e}_1=\{1,0,0\}; \quad \mathbf{e}_2=\{0,1,0\}; \quad \mathbf{e}_3=\{0,0,1\};$$

Se ubican los sistemas de referencia locales en la base de cada eslabón del robot como se muestra en la figura 5.5. Se observa que, en este caso, los sistemas de referencia definidos coinciden con la base canónica, la cual se considera como el sistema de referencia inercial fijo, es decir:

$$\mathbf{k}_j, \quad \mathbf{h}_j = \mathbf{g}_j = \mathbf{f}_j = \mathbf{e}_j, \quad j= 1,2,3.$$

Tabla 5.4 Cuaterniones asociados a los eslabones y sus bases locales			
Número de Eslabón	Configuración no deformada	Configuración deformada	
Eslabón 5 (pinza)		$r5=\{r50,0,0,r50\}$	
Eslabón 4		$r4=\{r40,r41,0,0\}$	
Eslabón 3	$Q=\{Q_0,Q_1,0,0\}$	$r3=\{r30,r31,0,0\}$	
Eslabón 2	$P=\{P_0,P_1,0,0\}$	$r2=\{r20,r21,0,0\}$	
Eslabón 1		$r1=\{r10,0,0,r13\}$	

De la figura mostrada en la tabla 5.4, se deben rotar los eslabones 2 y 3 a una posición angular arbitraria, considerando que los eslabones tienen los siguientes ejes de rotación, ver tabla 5.5. Con estos ejes definidos, se puede definir la forma que tendrán los Cuaterniones para rotar los eslabones 2 y 3, ver tabla 5.4.

Tabla 5.5 Ejes de rotación	
Eslabón	Eje de rotación
1	$e_3$
2	$f_1=e_1$
3	$g_1=e_1$
4	$h_1=e_1$
5	$k_3=e_3$

**Eslabón 2.** Se considera el siguiente parámetro que caracteriza a la primera rotación:

$$P = \{P_0, P_1, P_2, P_3\} \quad \rightarrow \quad P_V = \{P_1, P_2, P_3\}, \quad |P| = 1.$$

En este caso se tiene que si  $\varpi \in \mathfrak{R}^3$  es el eje de rotación, entonces,

$$\begin{aligned} P_0 &= \cos(\varphi_1/2) \\ P_V &= \text{sen}(\varphi_1/2) * \varpi \\ &= \text{sen}(\varphi_1/2) * f_1 \\ &= \text{sen}(\varphi_1/2) * e_1 \\ &= \text{sen}(\varphi_1/2) * \{1,0,0\} \\ P_V &= \{P_1,0,0\} \end{aligned}$$

De lo anterior, se obtiene la forma local del parámetro P, que es:

$$P = \{P_0, P_1, 0, 0\}, \text{ donde;} \quad (5.8)$$

$$\begin{aligned} P_0 &= \cos(\varphi_1/2) \\ P_V &= \text{sen}(\varphi_1/2) \end{aligned}$$

Siendo  $\varphi_1$  el ángulo de rotación del eslabón 2.

**Eslabón 3.-** El parámetro que caracterizará la segunda rotación:

$$Q = \{Q_0, Q_1, Q_2, Q_3\} \quad \rightarrow \quad Q_V = \{Q_1, Q_2, Q_3\}, \quad |Q| = 1.$$

En este caso se tiene que si  $\varpi \in \mathfrak{R}^3$  es el eje de rotación, entonces,

$$\begin{aligned} Q_0 &= \cos(\varphi_2/2) \\ Q_V &= \text{sen}(\varphi_2/2) * \varpi \\ &= \text{sen}(\varphi_2/2) * g_1 \\ &= \text{sen}(\varphi_2/2) * e_1 \\ &= \text{sen}(\varphi_2/2) * \{1,0,0\} \\ Q_V &= \{Q_1, 0, 0\} \end{aligned}$$

De lo anterior, se obtiene la forma local del parámetro Q, que es:

$$Q = \{Q_0, Q_1, 0, 0\}, \text{ donde;} \quad (5.9)$$

$$\begin{aligned} Q_0 &= \cos(\varphi_2/2) \\ Q_V &= \text{sen}(\varphi_2/2) \end{aligned}$$

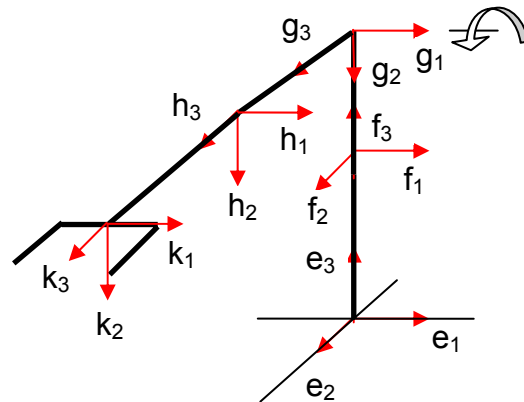
Siendo  $\varphi_2$  el ángulo de rotación del eslabón 3.

Con los Cuaterniones anteriores se puede realizar las rotaciones necesarias para determinar la configuración no deformada, se debe observar que si los ejes de rotación cambian su forma, también cambiará la forma de los Cuaterniones dados.

**V.4 APLICACIÓN DE LA SECUENCIA DE ROTACIONES PARA DETERMINAR LA CONFIGURACIÓN NO DEFORMADA.**

Rotación 2.- Se rota el tercer eslabón con el Cuaternión Q. Esto, se realiza rotando las bases locales  $g_j$ ,  $h_j$  y  $k_j$  (ver figura 5.3).

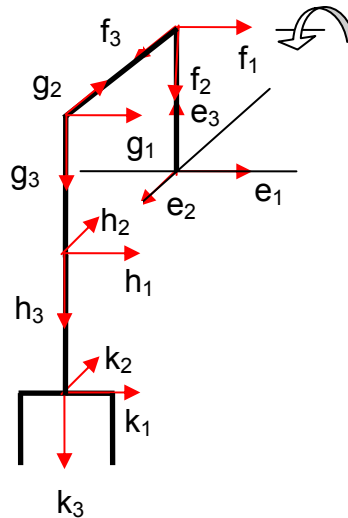
$$\begin{aligned}
 g_j^I &= \rho(Q, g_j) && \text{(Rotación del tercer eslabón con Cuaternión Q), } j = 1,2,3. \\
 h_j^I &= \rho(Q, h_j) && \text{(Rotación del cuarto eslabón con Cuaternión Q), } j = 1,2,3. \\
 k_j^I &= \rho(Q, k_j) && \text{(Rotación del quinto eslabón con Cuaternión Q), } j = 1,2,3.
 \end{aligned}
 \tag{5.10}$$



**Figura 5.3** Rotación del tercer eslabón en la configuración no deformada base  $g_j$ ,  $j = 1,2,3$ .

Rotación 3.- Se rota el segundo eslabón con el Cuaternión P. Esto, se giran las bases locales  $f_j$ ,  $g_j$  y  $h_j$  (ver figura 5.4).

$$\begin{aligned}
 f_j^I &= \rho(P, f_j) && \text{(Rotación del tercer eslabón con Cuaternión P), } j = 1,2,3. \\
 g_j^II &= \rho(P, g_j^I) && \text{(Rotación del tercer eslabón con Cuaternión P), } j = 1,2,3. \\
 h_j^II &= \rho(P, h_j^I) && \text{(Rotación del cuarto eslabón con Cuaternión P), } j = 1,2,3. \\
 k_j^II &= \rho(P, k_j^I) && \text{(Rotación del quinto eslabón con Cuaternión P), } j = 1,2,3.
 \end{aligned}
 \tag{5.11}$$



**Figura 5.4** Rotación del segundo eslabón en la configuración no deformada base  $f_j$ ,  $j = 1,2,3$ .

Como  $g_j^I = \rho(Q, g_j)$ ,  $h_j^I = \rho(Q, h_j)$  y  $k_j^I = \rho(Q, k_j)$  se tiene que:

$$\begin{aligned} g_j^{II} &= \rho(P, \rho(Q, g_j)) = \rho((P * Q), g_j) \\ h_j^{II} &= \rho(P, \rho(Q, h_j)) = \rho((P * Q), h_j) \\ k_j^{II} &= \rho(P, \rho(Q, k_j)) = \rho((P * Q), k_j) \end{aligned} \tag{5.12}$$

### V.5 VECTORES QUE DEFINEN LOS ESLABONES Y POSICIÓN DEL ELEMENTO TERMINAL EN LA CONFIGURACIÓN NO DEFORMADA.

De la figura 5.4 y 5.6, se pueden determinar los vectores que definen los eslabones del manipulador y el vector de posición del elemento terminal (punto p). Para calcular lo anterior, se reemplaza temporalmente las longitudes de la siguiente forma:  $L_1=d_1$ ,  $L_2=a_2$ ,  $L_3=a_3$ ,  $L_4=d_5$ , (Aquí se considera la longitud del gripper  $L_5=G$ ).

$$b_1 = L_1 * e_3 \tag{Eslabón 1} \tag{5.13}$$

$$b_2 = L_2 * f_3^I = L_2 * \rho(P * f_3) = L_2 * (P * e_3 * \bar{P}) \tag{Eslabón 2} \tag{5.14}$$

$$b_3 = L_3 * g_3^{II} = L_3 * \rho((P * Q), g_3) = L_3 * ((P * Q) * e_3 * \overline{P * Q}) \tag{Eslabón 3} \tag{5.15}$$

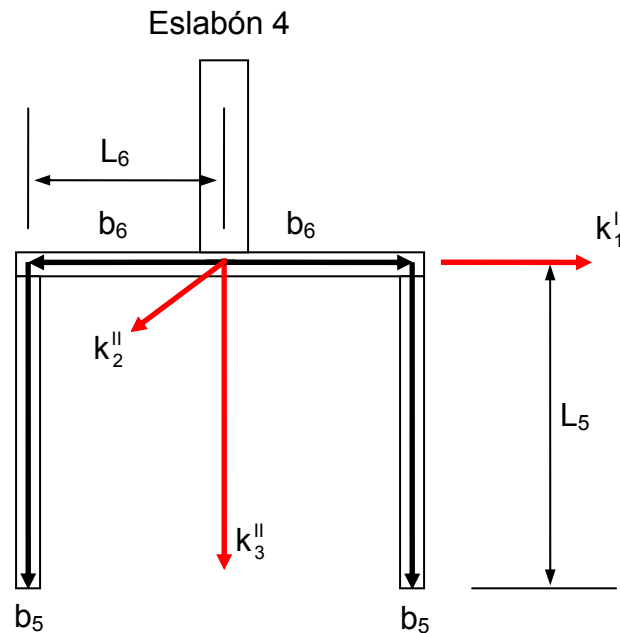
$$R_p = b_1 + b_2 + b_3 \tag{Posición del elemento terminal} \tag{5.16}$$

De la figura 5.4, se puede observar que la base local  $k_j^{II}$  es la que representa la pinza o eslabón 5 del manipulador. En la misma figura, se puede ver que son necesarios

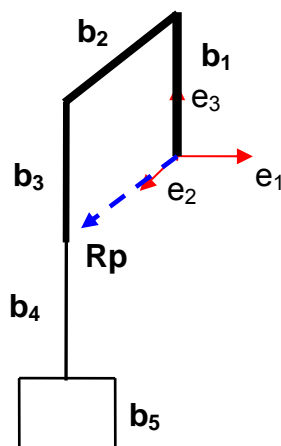
cuando menos dos vectores para definir y dibujar dicha pinza. Un vector es paralelo y el otro perpendicular a  $b_4$ , como se muestra en la figura 5.5. De lo anterior, se tiene que los vectores de la pinza son los siguientes:

$$b_5 = L_5 * k_3^{\parallel} = L_5 * \rho((P * Q), k_3) = ((P * Q) * e_3 * \overline{P * Q}) \quad (5.17)$$

$$b_6 = L_6 * k_1^{\parallel} = L_6 * \rho((P * Q), k_1) = ((P * Q) * e_1 * \overline{P * Q}) \quad (5.18)$$



**Figura 5.5** Vectores que representan la pinza del robot Mitsubishi®.



**Figura 5.6** Vector posición del elemento terminal en la configuración no deformada.

Las ecuaciones optimizadas de la configuración no deformada son las siguientes: La simplificación se realizó, utilizando el Software de Matemática<sup>®</sup> 4.0.

$$\begin{aligned} b_1 &= \{0, 0, L_1\} \\ b_2 &= \{0, -2 A_1 L_2, -A_2 L_2\} \\ b_3 &= \{0, 2 A_2 A_3 L_3 + 2 A_1 A_4 L_3, -4 A_1 A_3 L_3 + A_2 A_4 L_3\} \\ b_4 &= \{0, 2 A_2 A_3 L_4 + 2 A_1 A_4 L_4, -4 A_1 A_3 L_4 + A_2 A_4 L_4\} \\ b_5 &= \{0, 2 A_2 A_3 L_5 + 2 A_1 A_4 L_5, -4 A_1 A_2 L_5 + A_2 A_4 L_5\} \\ b_6 &= \{L_6, 0, 0\} \end{aligned}$$

$$R_p = \{0, -2 A_1 L_2 + 2 A_2 A_3 L_3 + 2 A_1 A_4 L_3, L_1 - A_2 L_2 - 4 A_1 A_3 L_3 + A_2 A_4 L_3\}$$

$$\begin{aligned} A_1 &= P_0 P_1 \\ A_2 &= 1 - 2 P_0^2 \\ A_3 &= Q_0 Q_1 \\ A_4 &= 1 - 2 Q_0^2 \end{aligned}$$

$$\begin{aligned} L_1 &= 0.300, \quad L_2 = 0.250, \quad L_3 = 0.160, \\ L_4 &= 0.072, \quad L_5 = 0.075, \quad L_6 = L_5/2 \end{aligned} \quad (5.18a)$$

En la figura 5.6a se muestra el robot Mitsubishi<sup>®</sup> en la posición de Home, incluyendo la orientación de la pinza. Para llevarlo a esa posición fue necesario plantear la cinemática inversa, para ello se implementó el siguiente sistema de ecuaciones para el plano Y, Z.

$$\begin{aligned} f_1 &= -2 A_1 L_2 + 2 A_2 A_3 L_3 + 2 A_1 A_4 L_3 = -0.2 & (\text{Eje Y}) \\ f_2 &= L_1 - A_2 L_2 - 4 A_1 A_3 L_3 + A_2 A_4 L_3 = 0.3 & (\text{Eje Z}) \\ f_3 &= P_0^2 + P_1^2 = 1 \\ f_4 &= Q_0^2 + Q_1^2 = 1 \end{aligned} \quad (5.18b)$$

Con las siguientes constantes:

$$\begin{aligned} A_1 &= P_0 P_1 \\ A_2 &= 1 - 2 P_0^2 \\ A_3 &= Q_0 Q_1 \\ A_4 &= 1 - 2 Q_0^2 \end{aligned}$$

Debido a que solo se presentaron rotaciones en la base  $e_1$  de cada eslabón, el eje X no se considera para este caso en particular. Los parámetros que resultaron fueron los siguientes, para alcanzar el punto del efector final en el espacio tridimensional: (0.0, -0.2, 0.3)

$$P_0 = 0.905331, \quad P_1 = 0.424706, \quad Q_0 = 0.446514, \quad Q_1 = 0.894777$$

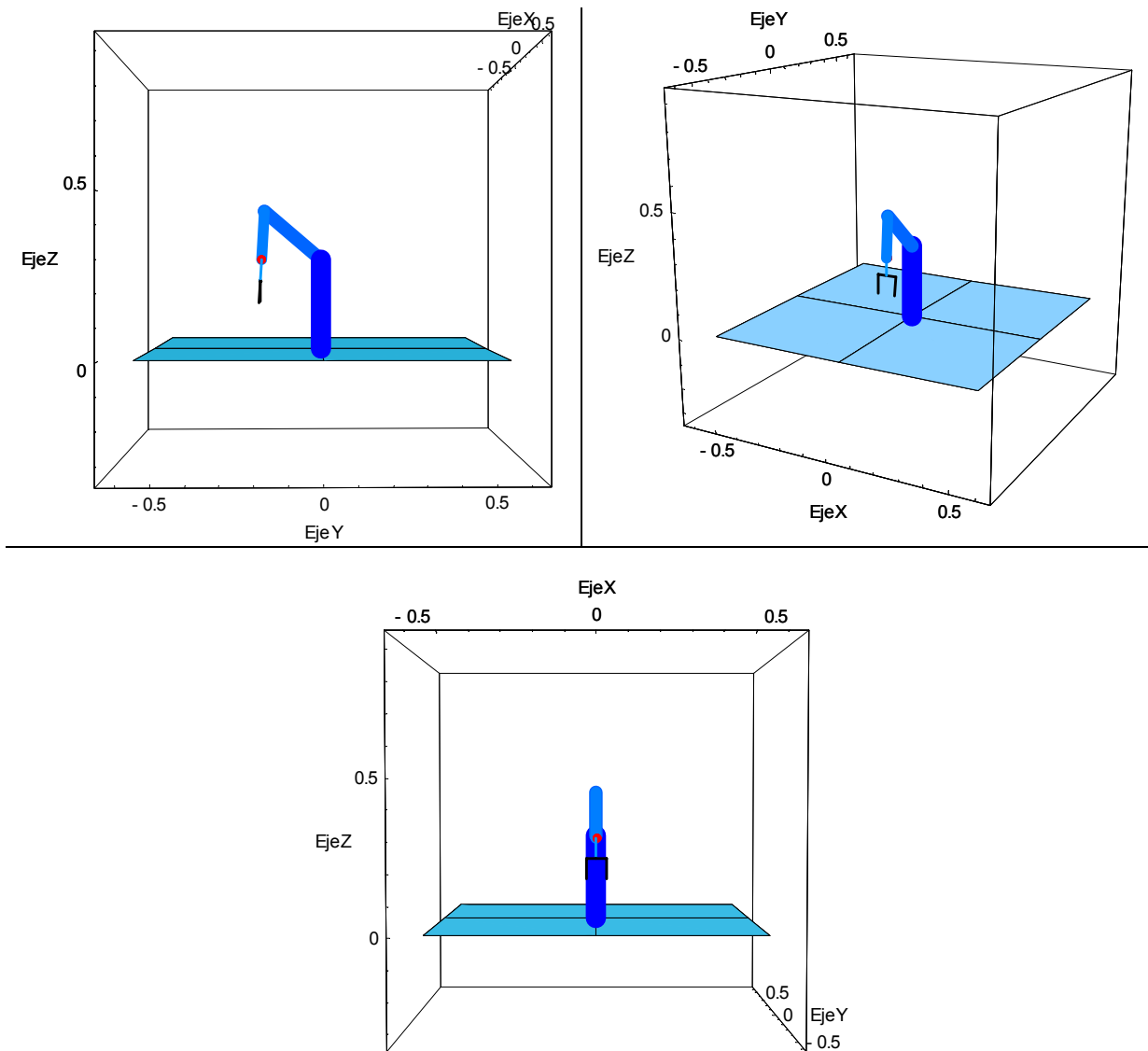
Los valores iniciales para resolver el sistema de ecuaciones (5.18b), utilizando el método de Newton-Raphson fueron:

$$iP_0 = 0.8, \quad iP_1 = 0.5, \quad iQ_0 = 0.5, \quad iQ_1 = 0.8$$

La orientación del órgano terminal, en este caso depende de la orientación del eslabón 3. Es decir, el eslabón de la pinza rotó la suma de las rotaciones de los eslabones 2 y 3,



por lo tanto, quedó como se muestra en la figura 5.6a, con ángulos de rotación igual a cero, para la pinza y su eslabón.



**Figura 5.6a** Posición no deformada (Home) del robot Mitsubishi®. Diferentes vistas.

## V.6 CONFIGURACIÓN DEFORMADA

Ahora, se modelará la configuración deformada, iniciando las rotaciones en el elemento Terminal y finalizando en el primer elemento, ya que es la secuencia más sencilla. Secuencia de rotaciones:

**Rotación 1.-** Rota el eslabón 5 con el quaternion  $r_5$  partiendo de la configuración no deformada, ver tabla 5.4 y figura 5.4. y para observar el eslabón (pinza) rotada, ver figura 5.7.

**Rotación 2.-** Rotan los eslabones 4 y 5 con el quaternion r4, ver figura 5.8.

**Rotación 3.-** Rotan los eslabones 3, 4 y 5 con el quaternion r3, ver figura 5.9.

**Rotación 4.-** Rotan los eslabones 2, 3, 4 y 5 con el quaternion r2, ver figura 5.10.

**Rotación 5.-** Rotan los eslabones 1, 2, 3, 4 y 5 con el quaternion r1, ver figura 5.11.

De la figura 5.4, se puede observar que los ejes de rotación de los eslabones se conservan como en la configuración original (ver tabla 5.4), excepto para el quinto eslabón, cuyo eje de rotación es  $k_3$ . Por lo anterior, se pueden definir los quaterniones para rotar a la posición deformada como se muestra a continuación (ver tabla 5.4):

$$\text{Eslabón 1: } r1 = \{r1_0, r1_1, r1_2, r1_3\} \quad \rightarrow \quad r1_v = \{r1_1, r1_2, r1_3\}, \quad |r1| = 1.$$

La forma del quaternion r1 es;

$$\begin{aligned} r1_0 &= \cos(\theta_1/2) \\ r1_v &= \text{sen}(\theta_1/2) * e_3 = \text{sen}(\theta_1/2) * \{0, 0, 1\} \\ r1_v &= \{0, 0, r1_3\} \end{aligned}$$

De lo anterior, se obtiene la forma local del parámetro r1, que es:

$$r1 = \{r1_0, 0, 0, r1_3\}, \text{ donde;} \quad (5.19)$$

$$\begin{aligned} r1_0 &= \cos(\theta_1/2) \\ r1_v &= \text{sen}(\theta_1/2) \end{aligned}$$

Siendo  $\theta_1$  el ángulo de rotación del eslabón 1 en la configuración deformada.

$$\text{Eslabón 2: } r2 = \{r2_0, r2_1, r2_2, r2_3\} \quad \rightarrow \quad r2_v = \{r2_1, r2_2, r2_3\}, \quad |r2| = 1.$$

La forma del quaternion r2 es;

$$\begin{aligned} r2_0 &= \cos(\theta_2/2) \\ r2_v &= \text{sen}(\theta_2/2) * e_1 = \text{sen}(\theta_2/2) * \{1, 0, 0\} \\ r2_v &= \{r2_1, 0, 0\} \end{aligned}$$

De lo anterior, se obtiene la forma local del parámetro r2, que es:

$$r2 = \{r2_0, r2_1, 0, 0\}, \text{ donde;} \quad (5.20)$$

$$\begin{aligned} r2_0 &= \cos(\theta_2/2) \\ r2_v &= \text{sen}(\theta_2/2) \end{aligned}$$

Siendo  $\theta_2$  el ángulo de rotación del eslabón 2 en la configuración deformada.

$$\text{Eslabón 3: } r_3 = \{r_{3_0}, r_{3_1}, r_{3_2}, r_{3_3}\} \quad \rightarrow \quad r_{3_V} = \{r_{3_1}, r_{3_2}, r_{3_3}\}, \quad |r_3| = 1.$$

La forma del quaternion  $r_3$  es;

$$\begin{aligned} r_{3_0} &= \cos(\theta_3/2) \\ r_{3_V} &= \text{sen}(\theta_3/2) * e_1 = \text{sen}(\theta_3/2) * \{1, 0, 0\} \\ r_{3_V} &= \{r_{3_1}, 0, 0\} \end{aligned}$$

De lo anterior, se obtiene la forma local del parámetro  $r_2$ , que es:

$$r_3 = \{r_{3_0}, r_{3_1}, 0, 0\}, \text{ donde;} \quad (5.21)$$

$$\begin{aligned} r_{3_0} &= \cos(\theta_3/2) \\ r_{3_V} &= \text{sen}(\theta_3/2) \end{aligned}$$

Siendo  $\theta_3$  el ángulo de rotación del eslabón 3 en la configuración deformada.

$$\text{Eslabón 4: } r_4 = \{r_{4_0}, r_{4_1}, r_{4_2}, r_{4_3}\} \quad \rightarrow \quad r_{4_V} = \{r_{4_1}, r_{4_2}, r_{4_3}\}, \quad |r_4| = 1.$$

La forma del quaternion  $r_4$  es;

$$\begin{aligned} r_{4_0} &= \cos(\theta_4/2) \\ r_{4_V} &= \text{sen}(\theta_4/2) * e_1 = \text{sen}(\theta_4/2) * \{1, 0, 0\} \\ r_{4_V} &= \{r_{4_1}, 0, 0\} \end{aligned}$$

De lo anterior, se obtiene la forma local del parámetro  $r_4$ , que es:

$$r_4 = \{r_{4_0}, r_{4_1}, 0, 0\}, \text{ donde;} \quad (5.22)$$

$$\begin{aligned} r_{4_0} &= \cos(\theta_4/2) \\ r_{4_V} &= \text{sen}(\theta_4/2) \end{aligned}$$

Siendo  $\theta_4$  el ángulo de rotación del eslabón 4 en la configuración deformada.

$$\text{Eslabón 5: } r_5 = \{r_{5_0}, r_{5_1}, r_{5_2}, r_{5_3}\} \quad \rightarrow \quad r_{5_V} = \{r_{5_1}, r_{5_2}, r_{5_3}\}, \quad |r_5| = 1.$$

La forma del quaternion  $r_5$  es;

$$\begin{aligned} r_{5_0} &= \cos(\theta_5/2) \\ r_{5_V} &= \text{sen}(\theta_5/2) * e_3 = \text{sen}(\theta_5/2) * \{0, 0, 1\} \\ r_{5_V} &= \{0, 0, r_{5_3}\} \end{aligned}$$

De lo anterior, se obtiene la forma local del parámetro  $r_5$ , que es:

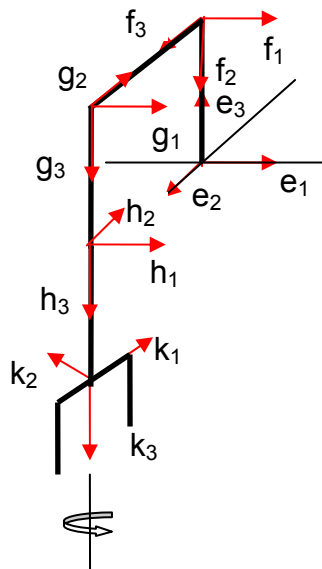
$$r_5 = \{r_{5_0}, 0, 0, r_{5_3}\}, \text{ donde;} \quad (5.23)$$

$$\begin{aligned} r_{5_0} &= \cos(\theta_5/2) \\ r_{5_V} &= \text{sen}(\theta_5/2) \end{aligned}$$

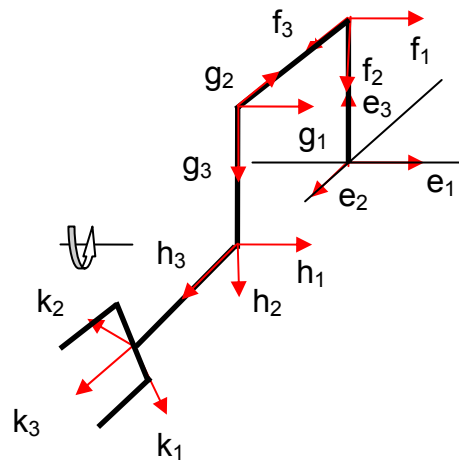
Siendo  $\theta_5$  el ángulo de rotación del eslabón 5 en la configuración deformada. Pero como el eje de rotación del eslabón 5 (pinza), se ve afectado por las rotaciones aplicadas en la configuración no deformada, siendo ahora  $k_3^{\parallel}$ , el quaternion  $r_5$ , será:

$$r_5^I = \text{Sin}(\theta_5/2) * k_3^{\parallel} = \text{Sin}(\theta_5/2) * \rho((P * Q), e_3) = \rho((P * Q), r_{5_v})$$

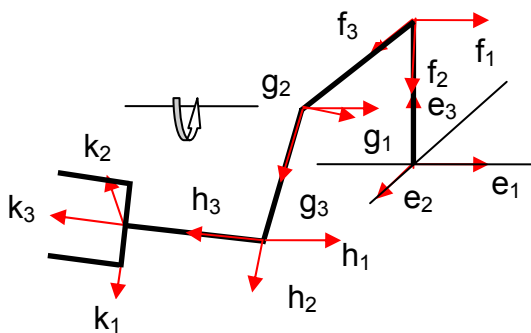
$$r_5^I = \rho((P * Q), r_5) \tag{5.24}$$



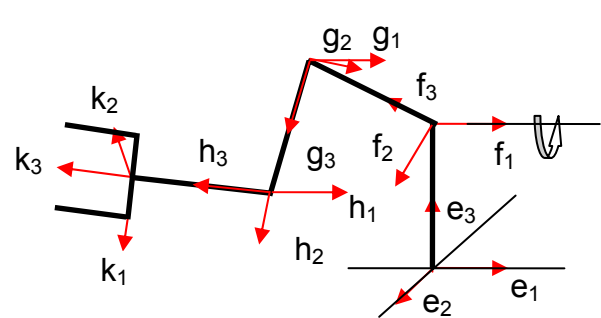
**Figura 5.7** Rotación del quinto eslabón (pinza).



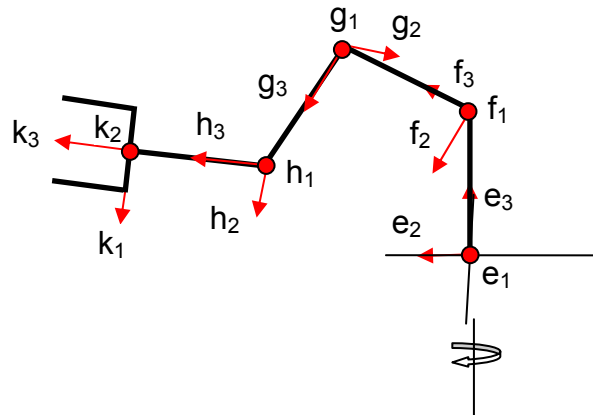
**Figura 5.8** Rotación del cuarto eslabón.



**Figura 5.9** Rotación del tercer eslabón.



**Figura 5.10** Rotación del segundo eslabón.



**Figura 5.11** Rotación del primer eslabón.

**V.7 APLICACIÓN DE LA SECUENCIA DE ROTACIONES PARA DETERMINAR LA CONFIGURACIÓN DEFORMADA.**

**Rotación 1.-** Rotación del eslabón 5 con el quaternion  $r_5$ , esta solamente afecta al eslabón 4. (figura 5.7).

$$k_j^{III} = \rho(r_5^I, k_j^{II}), \quad j = 1, 2, 3. \tag{5.25}$$

**Rotación 2.-** Rotación de los eslabones 4 y 5 con el quaternion  $r_4$ , éstas solamente afectan a las bases  $h_j$  y  $k_j$ . (figura 5.8).

$$\begin{aligned} h_j^{III} &= \rho(r_4, h_j^{II}), \\ k_j^{IV} &= \rho(r_4, k_j^{III}), \quad j = 1, 2, 3. \end{aligned} \tag{5.26}$$

**Rotación 3.-** Rotación de los eslabones 3, 4 y 5 con el quaternion  $r_3$ , éstas solamente afectan a las bases  $g_j$ ,  $h_j$  y  $k_j$ . (figura 5.9).

$$\begin{aligned} g_j^{III} &= \rho(r_3, g_j^{II}) \\ h_j^{IV} &= \rho(r_3, h_j^{III}), \\ k_j^{V} &= \rho(r_3, k_j^{IV}), \quad j = 1, 2, 3. \end{aligned} \tag{5.27}$$

**Rotación 4.-** Rotación de los eslabones 2, 3, 4 y 5 con el quaternion  $r_2$ , éstas solamente afectan a las bases  $f_j$ ,  $g_j$ ,  $h_j$  y  $k_j$ . (figura 5.10).

$$\begin{aligned} f_j^{II} &= \rho(r_2, f_j^I) \\ g_j^{IV} &= \rho(r_2, g_j^{III}) \\ h_j^{V} &= \rho(r_2, h_j^{IV}), \end{aligned}$$

$$k_j^{VI} = \rho(r_2, K_j^V), \quad j = 1, 2, 3. \quad (5.28)$$

**Rotación 5.-** Rotación de los eslabones 1, 2, 3, 4 y 5 con el cuaternión  $r_1$ , esta rotación afecta a todas bases  $e_j$ ,  $f_j$ ,  $g_j$ ,  $h_j$  y  $k_j$ . (figura 5.11).

$$\begin{aligned} e_j^I &= \rho(r_1, e_j) \\ f_j^{III} &= \rho(r_1, f_j^{II}) \\ g_j^V &= \rho(r_1, g_j^{IV}) \\ h_j^{VI} &= \rho(r_1, h_j^V), \\ k_j^{VII} &= \rho(r_1, K_j^{VI}), \quad j = 1, 2, 3. \end{aligned} \quad (5.29)$$

La forma final resultante de las bases locales de cada eslabón es la siguiente:

**Eslabón 1.-**

$$e_j^I = \rho(r_1, e_j) \quad (5.30)$$

**Eslabón 2.-**

$$\begin{aligned} f_j^{III} &= \rho(r_1, \rho(r_2, f_j^I)) = \rho(r_1, \rho(r_2, \rho(P, f_j))) = \rho(r_1 * r_2 * P, f_j) \\ &= \rho(r_1 * r_2 * P, e_j) \end{aligned} \quad (5.31)$$

**Eslabón 3.-**

$$\begin{aligned} g_j^V &= \rho(r_1, g_j^{IV}) = \rho(r_1, \rho(r_2, g_j^{III})) = \rho(r_1, \rho(r_2, \rho(r_3, g_j^{II}))) \\ &= \rho(r_1, \rho(r_2, \rho(r_3, \rho(P, g_j^I)))) = \rho(r_1, \rho(r_2, \rho(r_3, \rho(P, \rho(Q, g_j)))))) \\ &= \rho(r_1 * r_2 * r_3 * P * Q, e_j) \end{aligned} \quad (5.32)$$

**Eslabón 4.-**

$$\begin{aligned} h_j^{VI} &= \rho(r_1, h_j^V) = \rho(r_1, \rho(r_2, h_j^{IV})) = \rho(r_1, \rho(r_2, \rho(r_3, h_j^{III}))) = \rho(r_1, \rho(r_2, \rho(r_3, \rho(r_4, h_j^{II})))) \\ &= \rho(r_1, \rho(r_2, \rho(r_3, \rho(r_4, \rho(P, h_j^I)))))) = \rho(r_1, \rho(r_2, \rho(r_3, \rho(r_4, \rho(P, \rho(Q, h_j)))))) \\ &= \rho(r_1 * r_2 * r_3 * r_4 * P * Q, e_j) \end{aligned} \quad (5.33)$$

**Eslabón 5.-**

$$k_j^{VII} = \rho(r_1, K_j^{VI})$$

$$k_j^{VII} = \rho(r_1, k_j^{VI}) = \rho(r_1, \rho(r_2, k_j^V)) = \rho(r_1, \rho(r_2, \rho(r_3, k_j^{IV}))) = \rho(r_1, \rho(r_2, \rho(r_3, \rho(r_4, K_j^{III}))))$$

$$= \rho(r_1, \rho(r_2, \rho(r_3, \rho(r_4, K_j^{III})))) = \rho(r_1, \rho(r_2, \rho(r_3, \rho(r_4, \rho(r_5^I, K_j^II))))))$$

Pero como  $r_5^I = \rho((P * Q), r_5)$ , entonces:

$$\begin{aligned} &= \rho(r_1, \rho(r_2, \rho(r_3, \rho(r_4, \rho(\rho((P * Q), r_5), K_j^II)))))) \\ &= \rho(r_1, \rho(r_2, \rho(r_3, \rho(r_4, \rho(\rho((P * Q), r_5), \rho((P * Q), K_j)))))) \end{aligned}$$

Desarrollando la Transformación y aplicando algunas propiedades del álgebra de Cuaterniones, se tiene:

$$\begin{aligned} &= \rho(r_1, \rho(r_2, \rho(r_3, \rho(r_4, \rho(P * Q * r_5 * \overline{P * Q}, P * Q * K_j * \overline{P * Q})))))) \\ &= \rho(r_1, \rho(r_2, \rho(r_3, \rho(r_4, P * Q * r_5 * \overline{P * Q} * P * Q * K_j * \overline{P * Q} * P * Q * r_5 * \overline{P * Q})))))) \\ &= \rho(r_1, \rho(r_2, \rho(r_3, \rho(r_4, P * Q * r_5 * K_j * \overline{P * Q} * r_5)))))) \\ &= \rho(r_1, \rho(r_2, \rho(r_3, \rho(r_4, \rho(P, \rho(Q, \rho(r_5, k_j))))))) \\ &= \rho(r_1 * r_2 * r_3 * r_4 * P * Q * r_5, e_j) \end{aligned} \tag{5.34}$$

### V.8 VECTORES QUE DEFINEN LOS ESLABONES Y POSICIÓN DEL ELEMENTO TERMINAL EN LA CONFIGURACIÓN DEFORMADA

Siendo  $L_1, L_2, L_3, L_4$  y  $L_5$  las longitudes de los eslabones, se determina el vector posición  $R_p$  de la configuración deformada.

$$b_1 = L_1 * e_3^I \tag{Eslabón 1} \tag{5.35}$$

$$b_2 = L_2 * f_3^{III} \tag{Eslabón 2} \tag{5.36}$$

$$b_3 = L_3 * g_3^V \tag{Eslabón 3} \tag{5.37}$$

$$b_4 = L_4 * h_3^{VI} \tag{Eslabón 4} \tag{5.38}$$

$$b_5 = L_5 * k_3^{VII} \tag{Eslabón 5 Pinza} \tag{5.39}$$

$$b_6 = L_6 * k_1^{VII} \tag{Eslabón 5 Pinza} \tag{5.40}$$

$$R_p = b_1 + b_2 + b_3 + b_4 \tag{Posición del elemento terminal} \tag{5.41}$$

Las ecuaciones simplificadas y optimizadas de la configuración deformada son las siguientes: La simplificación se realizó, utilizando el Software de Matemática<sup>®</sup> 4.0.

$$\begin{aligned}
 b1 &= L1 ; \\
 b2 &= -5 B2 L2, - 2 A6 B2 L2, B1 L2 ; \\
 b3 &= -5 C1 L3, - 2 A6 C1 L3, C2 L3 ; \\
 b4 &= -A1 A5 B3 B5 L4 + 4 A12 A5 B4 B5 L4 + 4 A12 A5 B3 B6 L4 - 16 A11 A5 B4 B6 L4, \\
 & 2 A11 A6 B3 B5 L4 + 2 A12 A6 B4 B5 L4 + 2 A12 A6 B3 B6 L4 - 8 A11 A6 B4 B6 L4, \\
 & -A1 A6 B5 L4 + 4 A11 B4 B5 L4 + 4 A11 B3 B6 L4 + 4 A12 B4 B6 L4 ; \\
 b5 &= -A1 A5 B3 B5 L5 + 4 A12 A5 B4 B5 L5 + 4 A12 A5 B3 B6 L5 - 16 A11 A5 B4 B6 L5, \\
 & 2 A11 A6 B3 B5 L5 + 2 A12 A6 B4 B5 L5 + 2 A12 A6 B3 B6 L5 - 8 A11 A6 B4 B6 L5, \\
 & -A1 A6 B5 L5 + 4 A11 B4 B5 L5 + 4 A11 B3 B6 L5 + 4 A12 B4 B6 L5 ; \\
 b6 &= -A1 A6 L6 + 4 A12 A13 A5 B3 B5 L6 - 16 A11 A13 A5 B4 B5 L6 - 16 A11 A13 A5 B3 B6 L6 - 16 A12 A13 A5 B4 B6 L6, \\
 & - 2 A14 A5 L6 + 2 A12 A13 A6 B3 B5 L6 - 8 A11 A13 A6 B4 B5 L6 - 8 A11 A13 A6 B3 B6 L6 - 8 A12 A13 A6 B4 B6 L6, \\
 & 4 A11 A13 B3 B5 L6 + 4 A12 A13 B4 B5 L6 + 4 A12 A13 B3 B6 L6 - 16 A11 A13 B4 B6 L6 ; \\
 rp &= -5 A2 L2 + C1 L3, -A6 L2 + C1 L3, B1 L2 + C2 L3 ; \\
 e11 &= 2 A5, 0 ;
 \end{aligned}$$

$$\begin{aligned}
 A1 &= P0 P1; A2 = 1 - 2 P0^2; \\
 A3 &= Q0 Q1; A4 = 1 - 2 Q0^2; \\
 A5 &= r10 r13; A6 = 1 - 2 r10^2; \\
 A7 &= r20 r21; A8 = 1 - 2 r20^2; \\
 A9 &= r30 r31; A10 = 1 - 2 r30^2; \\
 A11 &= r40 r41; A12 = 1 - 2 r40^2; \\
 A13 &= r50 r53; A14 = 1 - 2 r50^2;
 \end{aligned}$$

$$\begin{aligned}
 B1 &= A2 A8 - 4 A1 A7; B2 = A2 A7 + A1 A8; \\
 B3 &= A10 A8 - 4 A7 A9; B4 = A10 A7 + A8 A9; \\
 B5 &= A2 A4 - 4 A1 A3; B6 = A2 A3 + A1 A4; \\
 C1 &= B4 B5 + B3 B6; C2 = B3 B5 - 4 B4 B6;
 \end{aligned}$$

$$\begin{aligned}
 L1 &= 0.300; L2 = 0.250; L3 = 0.160; L4 = 0.072; L5 = 0.075; L6 = N L4; \\
 z1 &= b1; z2 = b2; z3 = b3; z4 = b4; z5 = b5; z6 = b6; rx = rp;
 \end{aligned}$$

(5.42)

En la figura 5.12 se muestra el robot Mitsubishi® siguiendo una línea recta. La orientación de la pinza ha sido controlada para que esté paralela al eje Z y perpendicular a la trayectoria. El sistema de ecuaciones para el espacio X, Y, Z.

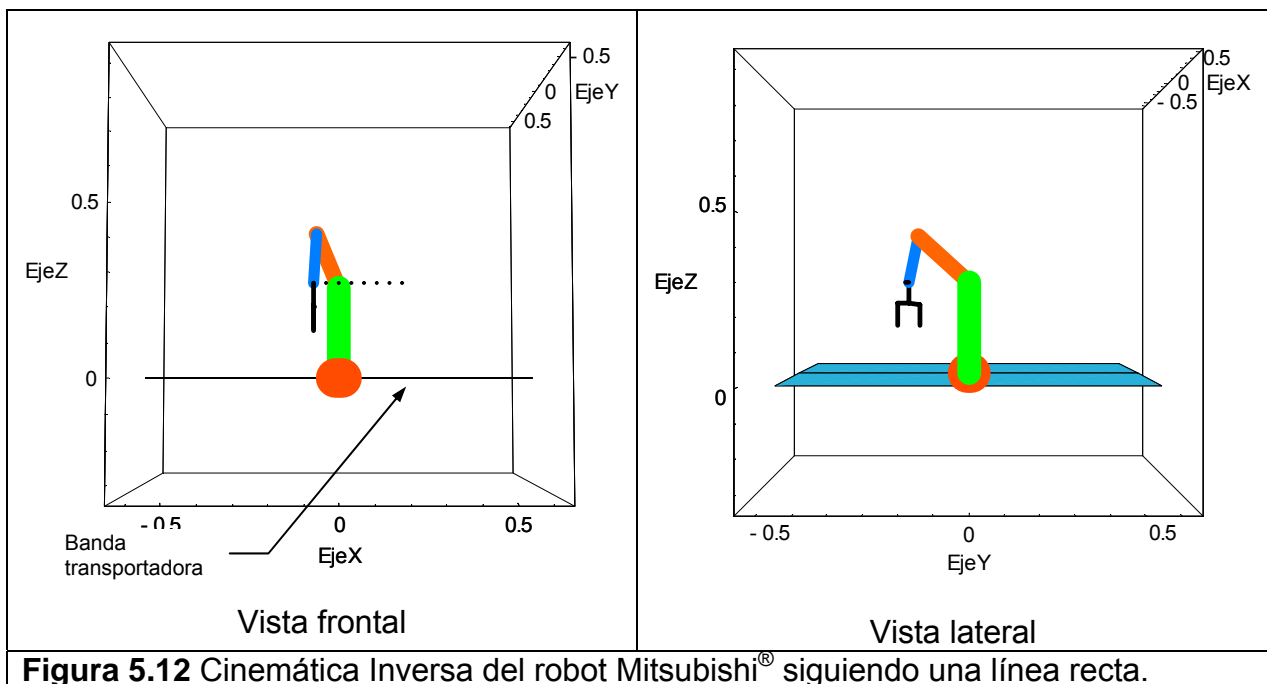
$$\begin{aligned}
 f1 &= -4 A5 (B2 L2 + C1 L3) = px && \text{(Eje X)} \\
 f2 &= -2 A6 (B2 L2 + C1 L3) = py && \text{(Eje Y)} \\
 f3 &= L1 + B1 L2 + C2 L3 = pz && \text{(Eje Z)} \\
 f4 &= r10^2 + r13^2 = 1 \\
 f5 &= r20^2 + r21^2 = 1 \\
 f5 &= r30^2 + r31^2 = 1
 \end{aligned}$$

(5.43)

Con las siguientes constantes:







**Figura 5.12** Cinemática Inversa del robot Mitsubishi® siguiendo una línea recta.

La orientación del órgano terminal conformado por el eslabón  $b_4$  y el eslabón (pinza), se determina de la siguiente manera: Debido a la configuración de la banda transportadora que se encuentra en el plano XZ (ver figura 5.12), el eslabón  $b_4$  debe orientarse verticalmente, es decir, paralelo al eje z, en este caso, paralelo a la base rotada  $e_{31}$ . Para lograrlo se aplica la ecuación (5.48), (ver código en Anexo C).

$$\text{cang} = \frac{b_1 \cdot b_4}{L_1 L_4}$$

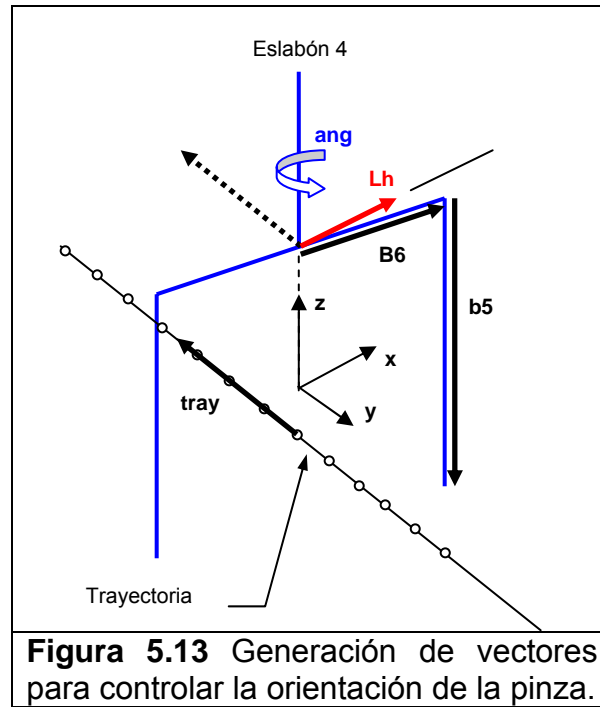
$$\text{If}[\text{cang} \neq 180^\circ, \text{ang1} = 180^\circ + \text{ArcCos}[\text{cang}], \text{ang1} = 0^\circ] \tag{5.48}$$

El ángulo  $\text{ang1}$  es la diferencia de rotación del eslabón  $b_4$ , para posicionarse perpendicularmente a la trayectoria y paralelamente al eslabón  $b_1$  (ver figura 5.13).

El eslabón 5 (pinza) también debe estar perpendicular a la trayectoria y paralelo al vector  $L_h$  de tal forma que la pinza capture la pieza en movimiento sobre la banda. Para lo anterior, se calculó el vector dirección unitario  $\text{tray}$  para cada punto de la trayectoria, ecuación (5.49 y 5.50).

$$\text{tray} = S_1 - S_0$$

$$\text{tray} = \frac{\text{tray}}{\sqrt{\text{tray}[[1]]^2 + \text{tray}[[2]]^2 + \text{tray}[[3]]^2}} \tag{5.49}$$



$$Lh = b_5 \cdot \text{tray}$$

$$\text{cosenoang} = \frac{b_6 \cdot Lh}{L_6 \sqrt{Lh[[1]]^2 + Lh[[2]]^2 + Lh[[3]]^2}} \quad (5.50)$$

$$\text{If}[\text{cosenoang} \neq 0^\circ, \text{ang} = \text{ArcCos}[\text{cosenoang}], \text{ang} = 90^\circ]$$

## V.9 SUMARIO

En este tema se aplicó la metodología para obtener los algoritmos de recuperación de tiempos aplicados a un robot dentro de un sistema de manufactura flexible en la etapa de simulación. El robot interceptor que en este caso es el robot industrial Mitsubishi® RV-1 fue modelado con el álgebra de cuaterniones. También se encontró el modelo cinemático utilizando cálculo matricial para comparar los resultados. Es importante mencionar las ventajas y desventajas que ofrecen los dos métodos: En esencia, los cuaterniones generan un sistema de ecuaciones (algebraicas) no lineales del tipo polinomial. Y el método matricial genera una matriz de 4x4 que contiene funciones trigonométricas de senos y cosenos. La ventaja que ofrecen las ecuaciones polinomiales es que no dependen de las funciones seno y coseno en forma directa y por lo tanto, se pueden simplificar y se puede aplicar un método interactivo como es el método numérico de Newton Raphson para resolverlas. Esto en el proceso de la cinemática inversa. A diferencia de tratar de resolver una matriz en donde se debe de acotar el sistema de 0 a  $2\pi$  rad y no se puede utilizar un método numérico en forma directa, además el método debe ser totalmente condicionado para no caer en soluciones de movimiento simétrico, es decir, como existen dos soluciones al menos, estas puedan hacer que el robot tenga otra configuración diferente o invertida a la esperada. Lo anterior, sería peligroso y crítico para el sistema y el robot.

Por otro lado, se estableció la orientación del efector final del robot para capturar el objeto de manera apropiada, esto es; sin impacto. Lo anterior se logró consideración una adecuación a la metodología propuesta para evitar el impacto entre las piezas y el efector final.

Lo anterior, demuestra que la metodología que genera algoritmos para recuperación de tiempos es satisfactoria al lograr generar trayectorias de intercepción múltiple de objetivos con impacto y sin impacto.

---

## CAPÍTULO VI

# ANÁLISIS DE RESULTADOS EXPERIMENTALES DE LOS EVENTOS REALIZADOS

### VI.1 GENERALIDADES

En este capítulo se estudian dos casos particulares de la aplicación de la metodología para recuperación de tiempos, El primero consiste en liberar dos esferas desde un dispositivo que controla su ángulo de disparo. Un robot interceptor prototipo de 2 grados de libertad las capturará en pleno vuelo con una pequeña red en sustitución del gripper.

En el segundo experimento particular se implementan los algoritmos de recuperación de tiempos obtenidos, utilizando la metodología propuesta, en la cinemática inversa del robot industrial Mitsubishi<sup>®</sup>, que capturará dos piezas que se mueven sobre la banda transportadora de un Sistema de Manufactura Flexible (FMS). Con un sistema de visión diseñado y desarrollado se obtendrá la información cinemática como son la velocidad y la distancia entre los centros de las piezas.

La originalidad de este evento radica en que se atrapan dos objetos en forma múltiple. La trayectoria TPS que genera la metodología captura en pleno vuelo las dos partículas. Con este estudio experimental se podrá validar la metodología. A nivel mundial solo se atrapan objetos únicos, es decir, uno por uno. En términos de eficiencia existe un criterio de ahorro de energía pero sobre todo de tiempo en la realización de esta tarea para el robot.

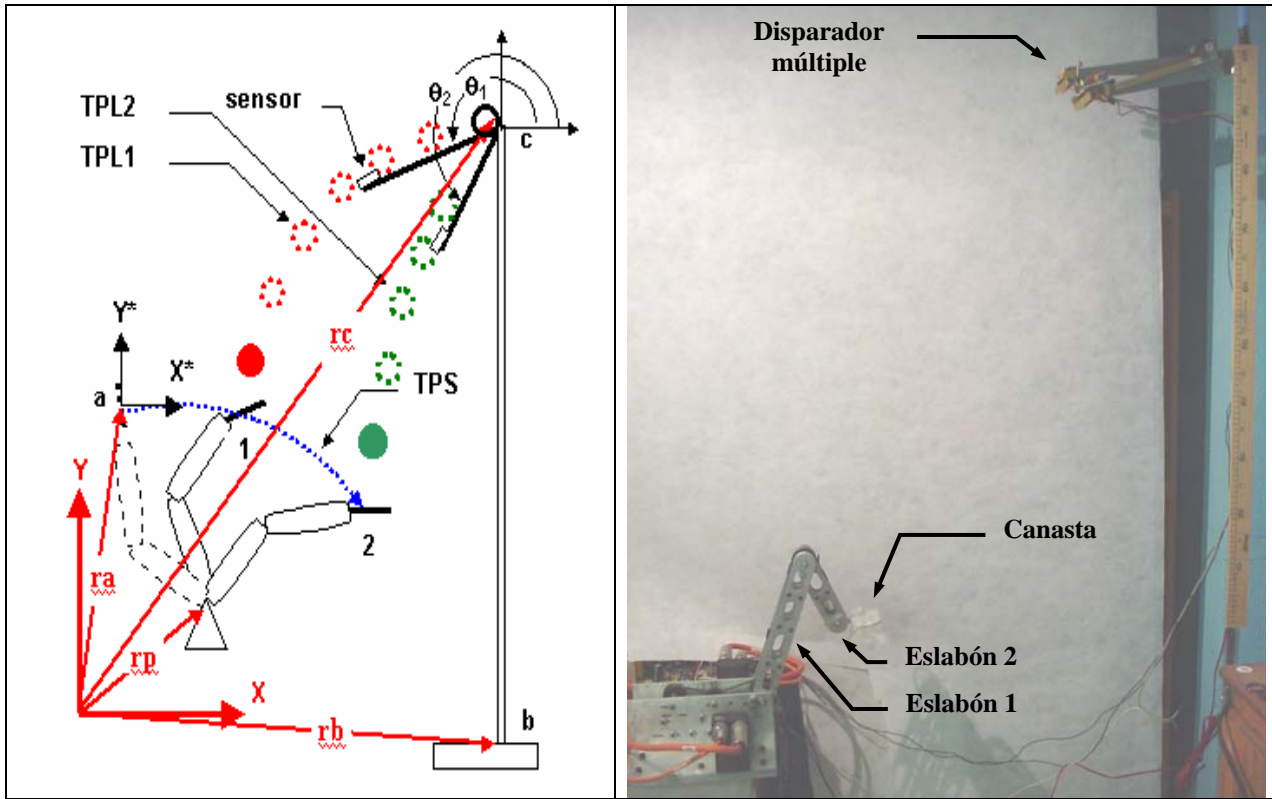
### VI.2 ANÁLISIS EXPERIMENTAL DE UN CASO PARTICULAR DE TIRO PARABÓLICO

En la figura 6.1a se muestra el esquema para realizar el evento experimental, el robot es de 2 grados de libertad y cuenta con una canasta que funge como gripper para capturar las esferas. Es importante mencionar, que la orientación de la canasta será controlada por medio de un control P (Proporcional) externo al sistema pero conectado al programa principal que integra a todo el sistema.

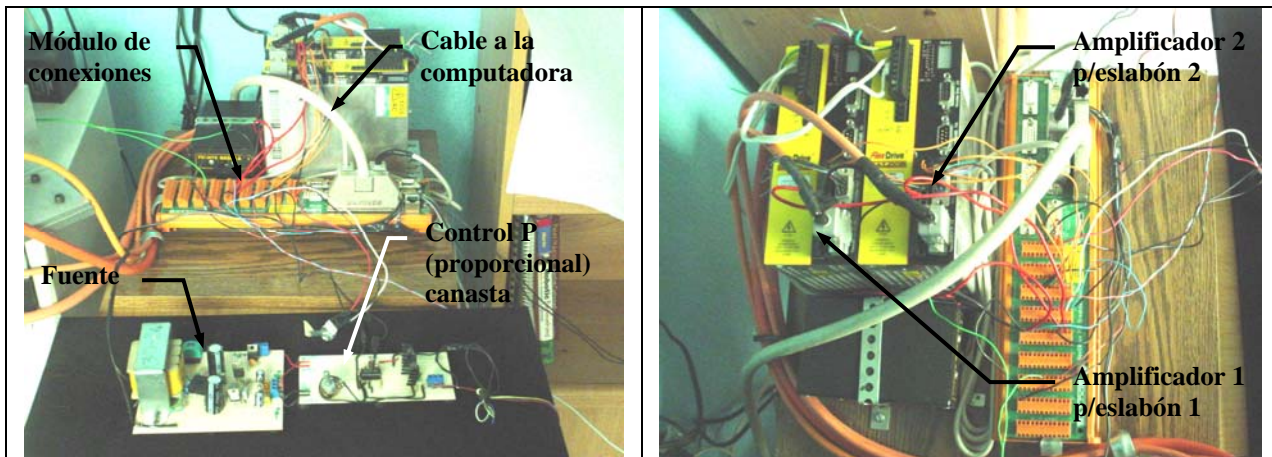
Con este experimento se pretende obtener las ganancias del controlador PIDVA que fueron necesarias para que los eslabones del robot interceptor generen las trayectorias que involucran posición y velocidad. Así como también, los parámetros cinemáticos como son; la posición y velocidad de cada motor. Las velocidades máximas a las cuales puede operar el robot y las condiciones extremas de operación.

Las esferas serán liberadas con una señal de +24 Vcc que activará a un par de selenoides que sujetan las esferas, de tal forma que el tiempo de liberación será el

mismo para las dos esferas. Los ángulos de disparo serán fijados arbitrariamente, ver figura 6.1a.



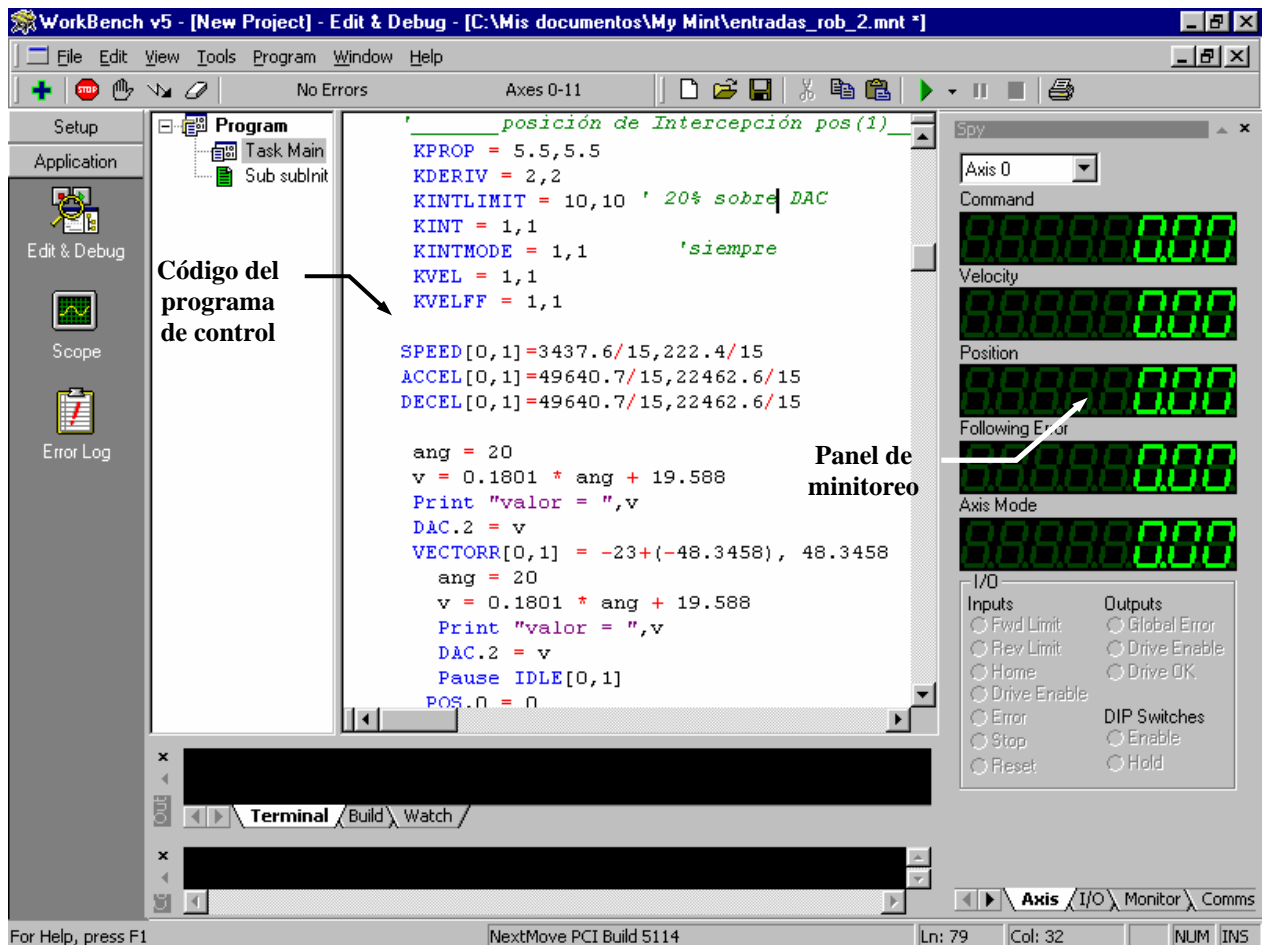
**Figura 6.1a** Esquema general de los parámetros del evento experimental y montaje.



**Figura 6.1b** Montaje del servo control PID para realizar la parte experimental.

La tabla 6.1a muestra el esquema general con los datos y las condiciones en las cuales se va a desarrollar el evento experimental de la figura 6.1a el diámetro de las esferas es de 2.5 cm. En la figura 6.1b se muestra el montaje del sistema de servo control que consta de dos amplificadores, un módulo de conexiones y un control P (proporcional). Los dos amplificadores reciben señales desde la computadora y las amplifica para

luego enviarlas a los motores que mueven a los eslabones uno y dos , respectivamente. El control p solo controla el efector final del robot (canasta). El programa que controla al robot se muestra en la figura 6.1c, éste consta de un panel de monitoreo que despliega información relacionada con el movimiento del robot, como: la posición, la velocidad, el error de seguimiento y el número de motor al que hace referencia. Además, permite cambiar las ganancias y los perfiles de velocidad y aceleración en el tiempo de ejecución del programa. Monitorea las entradas y salidas digitales, así como también las analógicas. En el código se pueden realizar operaciones algebraicas como son suma, resta, multiplicación, etc. Después de realizar una corrida la información la almacena en buffers que después son leídos para realizar las gráficas de posición, velocidad y el error de seguimiento.



**Figura 6.1c** Interfase gráfica de control PID para realizar la parte experimental.

En la tabla 6.1 se muestra las distancias y la configuración que se utilizaran para realizar la parte experimental, ver también figura 6.1a.

<b>rp(x,y)</b>	(0,0)
<b>rb(x,y)</b>	(0.80,-0.90)
<b>rc(x,y)</b>	(0.80,1.10)
<b>ra(x,y)</b>	( 0.15, 0.20)

El evento experimental que se propone resulta ser el mas crítico debido a que los algoritmos se prueban en el robot interceptor de 2 gdl donde las partículas libres TPL1 y TPL2 son liberadas a una altura menor a 1 metro, por lo tanto, la velocidad que se demanda en los actuadores es muy grande. Esta configuración garantiza que el efecto del rozamiento del aire y el efecto magnus de las esferas en movimiento no altere su trayectoria ideal en forma considerable.

### VI.3 CONDICIONES Y RESTRICCIONES DEL EXPERIMENTO

A continuación se presentan las restricciones físicas para realizar este evento experimental. Primero se toma la distancia entre el efector final (canasta) del robot y la salida de las esferas menor a un metro de altura, esto con el fin de evitar el efecto magnus y la fricción del aire sobre la trayectoria de las esferas al caer. Evidentemente, la aceleración y velocidades angulares que deben alcanzar los eslabones del robot interceptor será grande pero dentro de los límites de velocidad y aceleración nominal del sistema de control y el par máximo de los motores.

Esta situación será crítica y por ende se seleccionaron estas condiciones. En la tabla 6.2 se establecen las distancias a las cuales se encuentra la canasta del robot y las esferas, las esferas serán liberadas y el sensor de velocidad calculará la velocidad de salida de las esferas en milisegundos, ver figura 4.4 del capítulo IV.

**Tabla 6.2** Datos y condiciones iniciales del experimento

Esfera #1 TPL1			Esfera #2 TPL2		
Posición inicial:	$\theta_0 =$	200°	Posición inicial:	$\theta_0 =$	190°
	$X_0 =$	0.65 m		$X_1 =$	0.65 m
	$Y_0 =$	0.75 m		$Y_1 =$	0.80 m
Velocidad:	$V_0 =$	1.295m/s	Velocidad:	$V_1 =$	0.923m/s
Tiempo de retardo:	$t_1 =$	0.24 s	Tiempo de retardo:	$t_1 =$	0.24 s
Tiempo de intercepción 1:	$t_2 =$	0.26 s	Tiempo de intercepción 2:	$t_4 =$	0.34 s
Tiempo $\Delta t$	$\Delta t =$	0.02 s	Tiempo $\Delta t$	$\Delta t =$	0.02 s
Tiempo para atrapar	$t_3 =$	0.28 s	Tiempo para atrapar	$t_5 =$	0.36 s
Punto de inicio de TPSx	$ax =$	0.3 m	Punto de inicio de TPSx	$ax =$	0.3 m
Punto de inicio de TPSy	$ay =$	0.2 m	Punto de inicio de TPSy	$ay =$	0.2 m

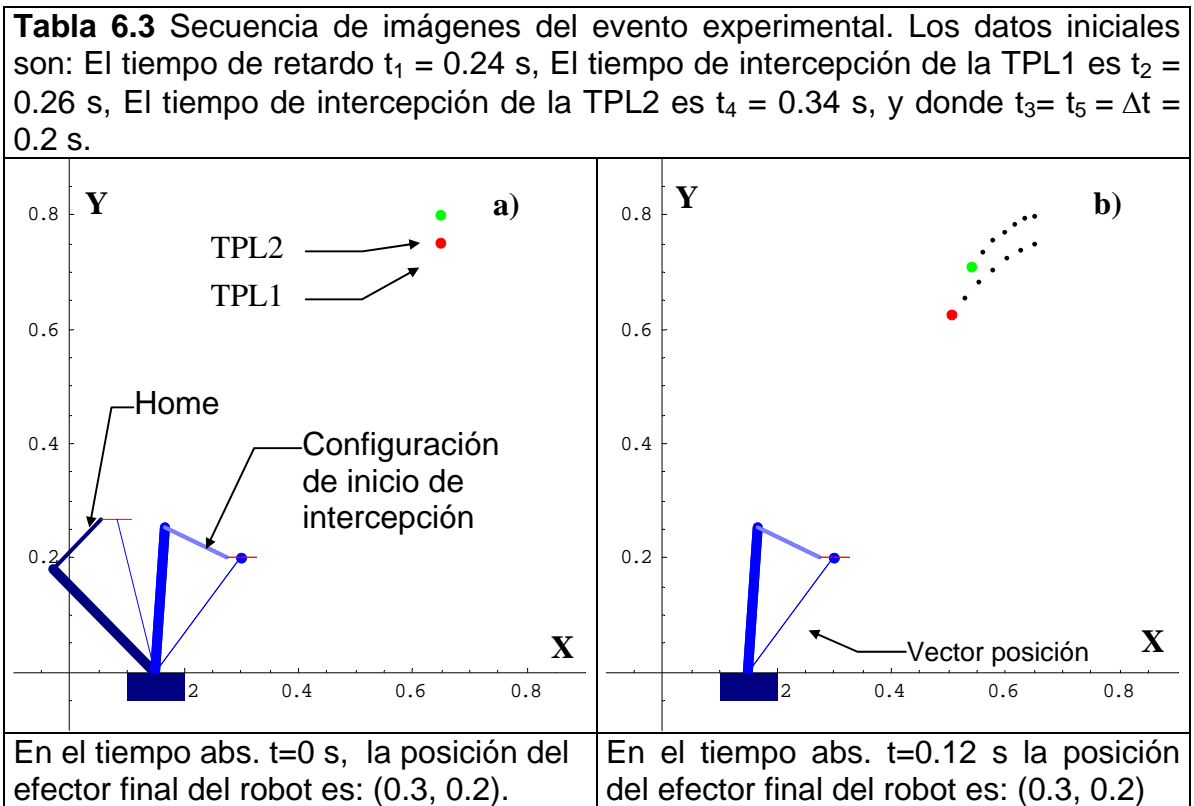
A continuación en la tabla 6.3 se presenta la secuencia de imágenes del evento de intercepción múltiple. Cabe mencionar que en cada imagen se presentan las posiciones

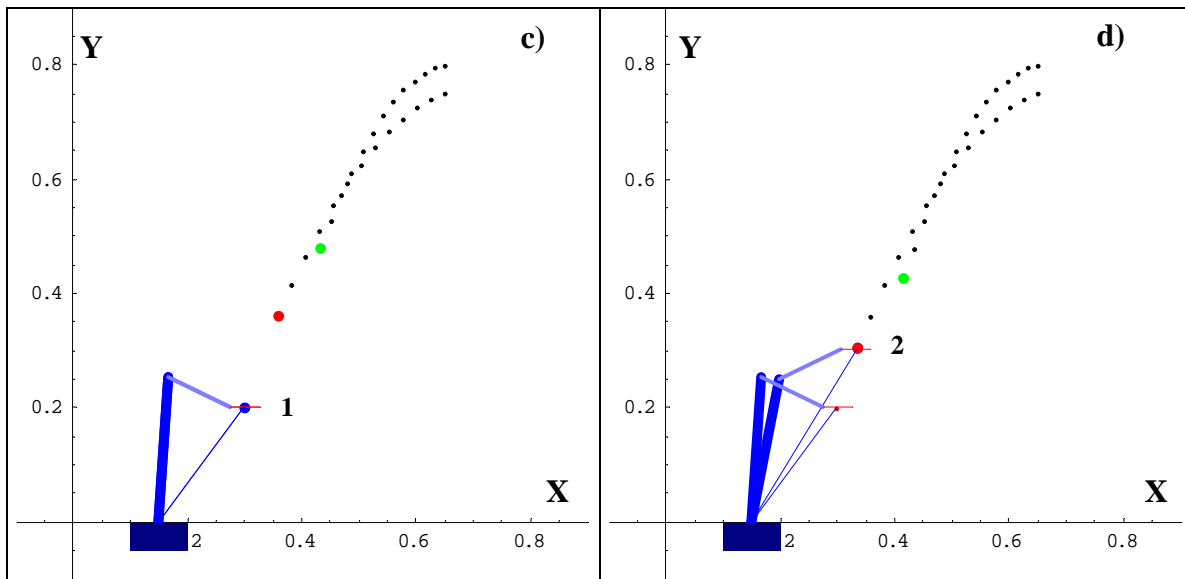


actual del robot y desde la cual el robot se movió, lo anterior con la idea de observar el desplazamiento incremental (coordenadas relativas).

El intervalo de tiempo es de  $\Delta t = 0.2$  s. La posición de la base del robot es  $(h,k) = (0.15,0)$ . El tipo de intercepción es sin impacto, es decir, la canasta del robot interceptor iguala la velocidad de cada esfera un instante  $\Delta t$  antes de capturarla.

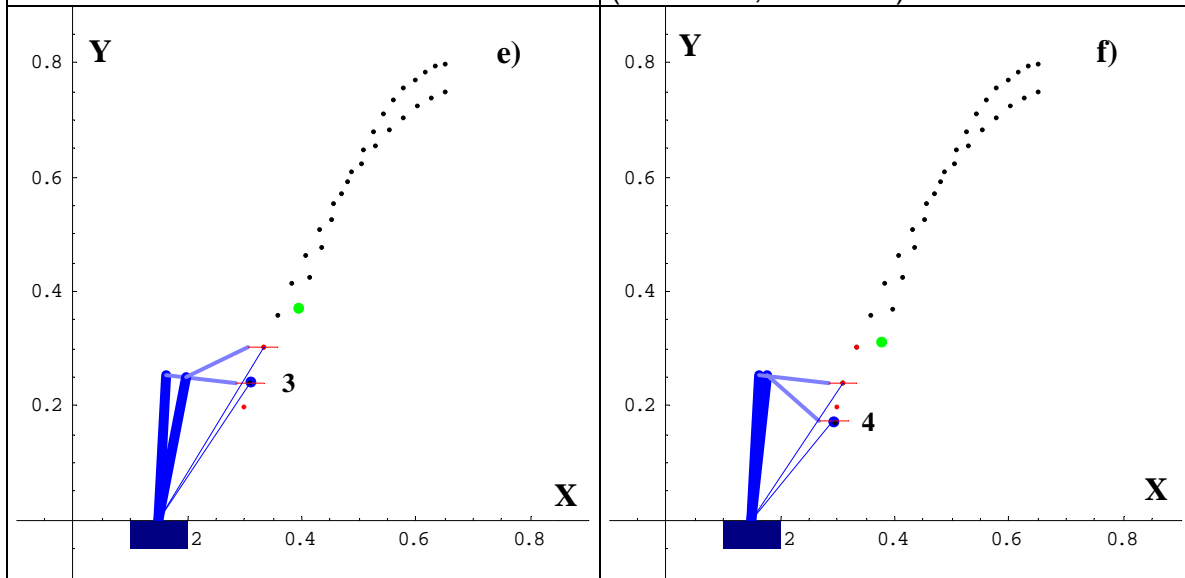
En la figura de la tabla 6.3a se muestra el robot en la posición de configuración de inicio de intercepción que se mide desde la posición de "home". El eslabón 3 es decir la canasta, se orienta paralelamente al eje horizontal (eje x), debido a la dirección de las esferas que caen libremente. En la Figura de la tabla 6.3b las esferas tienen la posición mostrada en el tiempo  $t=0.12$  s, después de ser liberadas. Además, se muestra el vector posición medido desde la base del robot al centro del efector final del robot. En el instante de tiempo anterior mencionado el robot todavía no se mueve. En la figura 6.3d de la misma tabla, el robot ya inició su movimiento e intercepta a la primera partícula TPL1 igualando su velocidad para evitar el impacto.





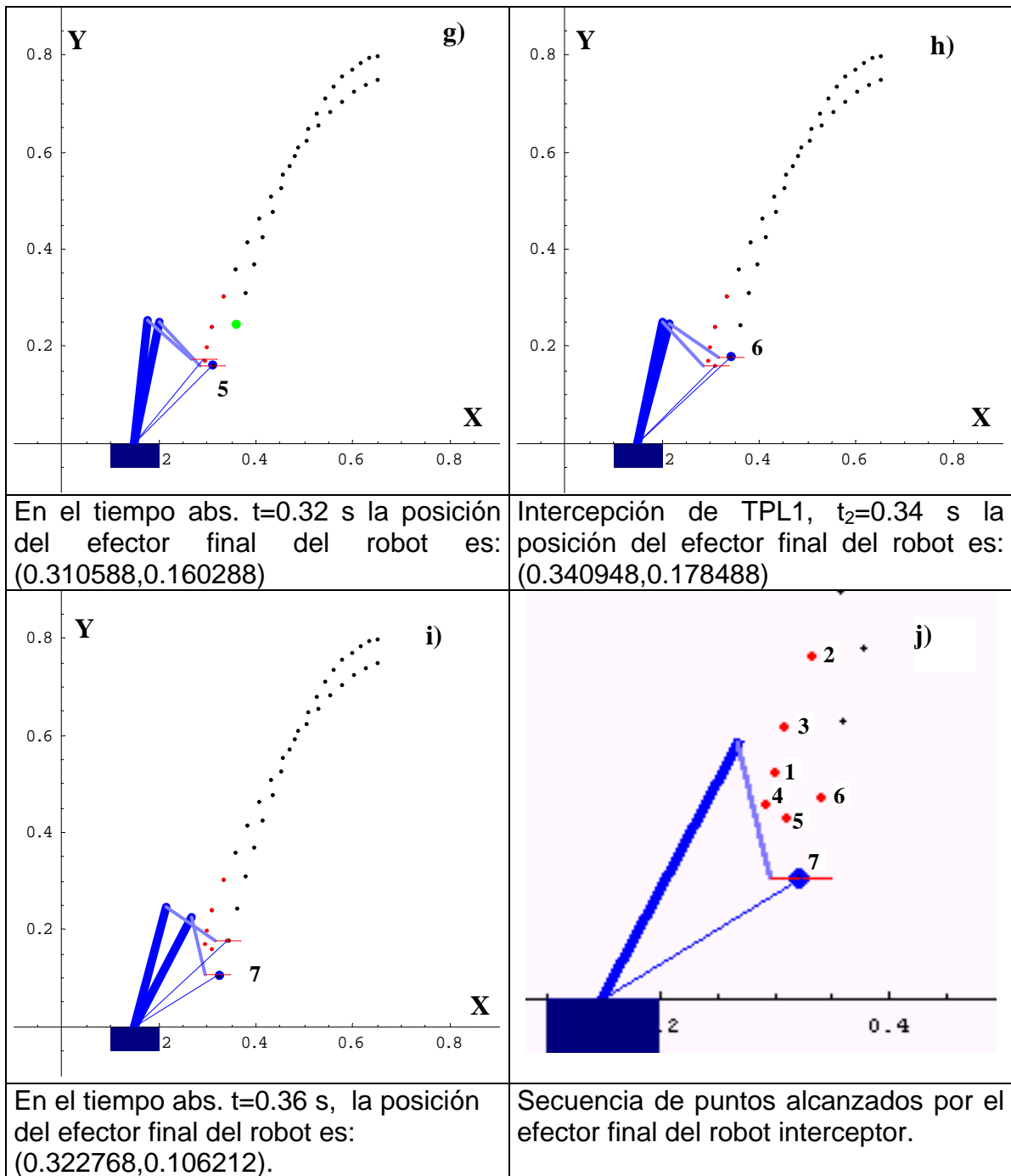
En el tiempo abs.  $t=0.24$  s la posición del efector final del robot es: (0.3,0.2)

Intercepción de TPL1,  $t_2=0.26$  s la posición del efector final del robot es: (0.333605,0.303264)



En el tiempo abs.  $t=0.28$  s, la posición del efector final del robot es: (0.309267,0.241461).

En el tiempo abs.  $t=0.30$  s la posición del efector final del robot es: (0.293664,0.172095)



En la figura de la tabla 6.3h el robot intercepta la segunda partícula TPL2, después iguala la velocidad de la partícula de tal forma que el impacto es nulo.

En la figura 6.2, se muestran los puntos por los cuales paso el efector final del robot interceptor mostrados en los incisos a al i) de la tabla 6.3a. Estos puntos han sido interpolados con la finalidad de tener la trayectoria unida por líneas. Es importante indicar que el punto con el número 2 es donde se intercepta la primera esfera TPL1, y en el punto número 6 se intercepta la partícula 2, TPL2.

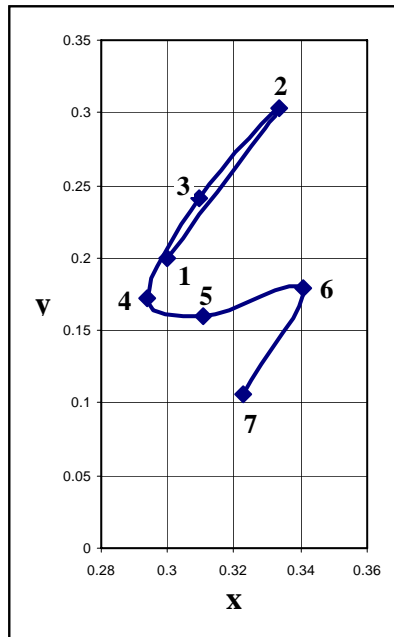


Figura 6.2 La trayectoria TPS

A continuación en la tabla 6.4 se presentan los resultados numéricos del evento de intercepción: la posición de “home” del robot tiene las coordenadas articulares  $(\theta_1(t), \theta_2(t), \theta_3(t)) = (135, -90, -45)$  grados, el signo negativo indica rotación en sentido horario.

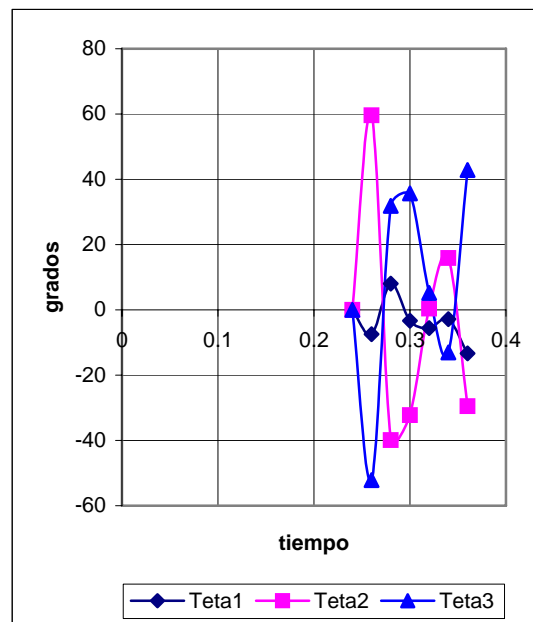
Tabla 6.4 Resultados numéricos de la simulación del evento de intercepción múltiple

#	Tiempo	Posición de TPS		Coordenadas Absolutas			Coordenadas Relativas (Gra)		
		X m	Y m	$\theta_1(t)$ Teta1	$\theta_2(t)$ Teta2	$\theta_3(t)$ Teta3	$\theta_1(t)$ Teta1	$\theta_2(t)$ Teta2	$\theta_3(t)$ Teta3
1	0	0.3	0.2	-48.3458	-23.222	71.5678	0	0	0
2	0.02	0.3	0.2	-48.3458	-23.222	71.5678	0	0	0
3	0.04	0.3	0.2	-48.3458	-23.222	71.5678	0	0	0
4	0.06	0.3	0.2	-48.3458	-23.222	71.5678	0	0	0
5	0.08	0.3	0.2	-48.3458	-23.222	71.5678	0	0	0
6	0.1	0.3	0.2	-48.3458	-23.222	71.5678	0	0	0
7	0.12	0.3	0.2	-48.3458	-23.222	71.5678	0	0	0
8	0.14	0.3	0.2	-48.3458	-23.222	71.5678	0	0	0
9	0.16	0.3	0.2	-48.3458	-23.222	71.5678	0	0	0
10	0.18	0.3	0.2	-48.3458	-23.222	71.5678	0	0	0
11	0.2	0.3	0.2	-48.3458	-23.222	71.5678	0	0	0
12	0.22	0.3	0.2	-48.3458	-23.222	71.5678	0	0	0
13	0.24	0.3	0.2	-48.3458	-23.222	71.5678	0	0	0
14	0.26	0.333605	0.303264	-55.7576	36.3736	19.384	-7.41171	59.5956	-52.1839
15	0.28	0.309267	0.241431	-47.6957	-3.55644	51.2522	8.06185	-39.93	31.8682
16	0.3	0.293664	0.172095	-51.0479	-35.8585	86.9064	-3.35221	-32.3021	35.6543
17	0.32	0.310588	0.160288	-56.6767	-35.4678	92.1445	-5.62873	0.390695	5.23803
18	0.34	0.340948	0.178488	-59.5142	-19.5723	79.0865	-2.83755	15.8956	-13.058
19	0.36	0.322768	0.106612	-72.8583	-49.0617	121.92	-13.3441	-29.4895	42.8336

En la misma tabla se considera todo evento completo, es decir, desde el tiempo  $t= 0$  s hasta el tiempo  $t= 0.36$  s, y en la posición # 14, el robot intercepta la partícula 1, TPL1. En la posición # 18 el robot intercepta la partícula 2, TPL2. Cabe mencionar, que en el tiempo  $t=0$  s, la posición de inicio de intercepción del robot interceptor es  $(\theta_1(t), \theta_2(t), \theta_3(t)) = (-48.3458, -23.222, 71.5678)$  medidos desde la posición de “home”, ver tabla 6.3 inciso a). Además, las coordenadas relativas son las que se van a programar en el controlador del robot.

#### VI.4 ANÁLISIS DE LAS COORDENADAS ARTICULARES DEL ROBOT INTERCEPTOR

En la figura 6.3 se presenta la gráfica de las coordenadas angulares del robot interceptor, el ángulo  $\theta_2(t)$  alcanza un desplazamiento angular de  $60^\circ$  y  $\theta_3(t)$  tiene el desplazamiento más pequeño que es de  $10^\circ$  como máximo.



**Figura 6.3** Evolución de las variables articulares en el tiempo del robot interceptor.

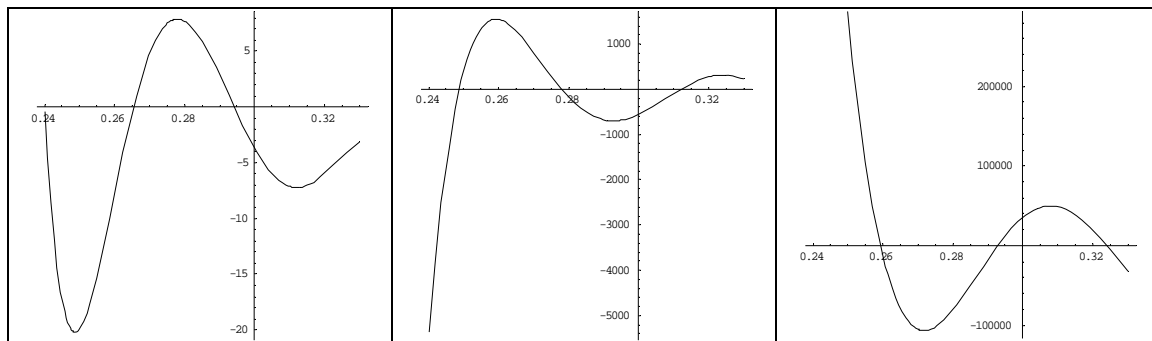
Las funciones que representan la evolución de las coordenadas en el tiempo son: para la posición, velocidad y aceleración angular:

$$\text{teta1}(t) = 3291937062 t^6 - 6055816972 t^5 + 4622585144 t^4 - 1873919383 t^3 + 425449402 t^2 - 51287017 t + 2564352 \quad (6.1)$$

$$\text{Vteta1}(t) = 19751622372 t^5 - 30279084860 t^4 + 18490340576 t^3 - 5621758149 t^2 + 850898804 t - 51287017 \quad (6.2)$$

$$A_{\theta 1}(t) = 98758111860 t^4 - 121116339440 t^3 + 55471021728 t^2 - 11243516298 t + 850898804 \quad (6.3)$$

En la figura 6.4 se muestran las gráficas de posición, velocidad y aceleración que se obtuvieron de las ecuaciones 6.1, 6.2 y 6.3, se puede observar que la aceleración angular del eslabón 1 del robot interceptor es grande comparada con las aceleraciones de los 2 eslabones restantes. Lo anterior, es debido al área de intercepción en donde puede moverse el robot, a la velocidad con que caen las esferas y sobre todo a la distancia relativa que existe entre ellas. Esta información es registrada en el controlador para ejecutar el movimiento del robot y se encuentra cerca de los límites de velocidad y aceleración a los cuales puede responder. Este experimento se realizó de tal forma que se utilizará la máxima velocidad y aceleración permitida por el controlador del robot.



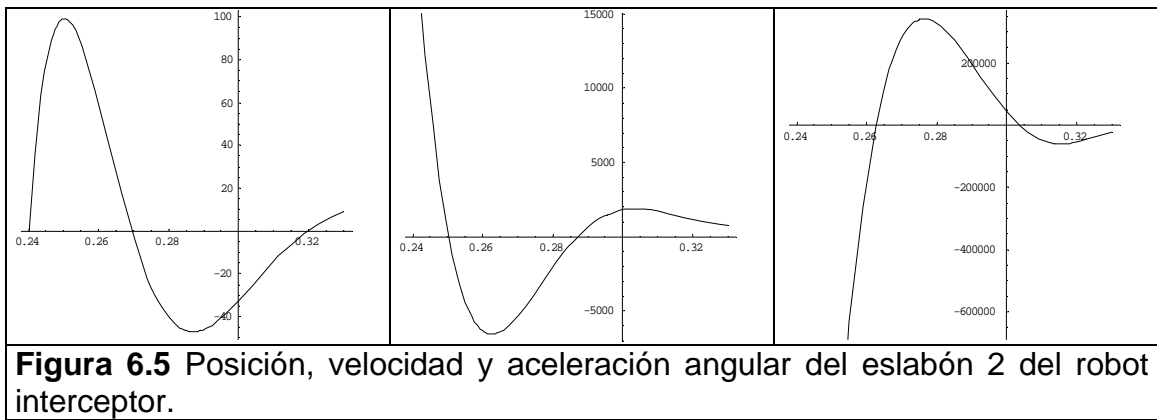
**Figura 6.4** Posición, velocidad y aceleración angular del eslabón 1 del robot interceptor.

$$\theta_2(t) = -9320405267 t^6 + 17228440385 t^5 - 13231098746 t^4 + 5402542348 t^3 - 1236723238 t^2 + 150447662 t - 7596518 \quad (6.4)$$

$$V_{\theta 2}(t) = -55922431602 t^5 + 86142201925 t^4 - 52924394984 t^3 + 16207627044 t^2 + 2473446476 t + 50447662 \quad (6.5)$$

$$A_{\theta 2}(t) = -279612158010 t^4 + 344568807700 t^3 - 158773184952 t^2 + 32415254088 t - 2473446476 \quad (6.6)$$

En la figura 6.4 se presentan las gráficas correspondientes a las ecuaciones 6.4 a la 6.6, respectivamente. Las trayectorias son suaves, es decir, se tiene un movimiento del eslabón que puede seguir el controlador sin dificultad, nuevamente la aceleración es grande y no es constante, como en el caso de la aceleración de la gravedad que afecta a las esferas en su movimiento de caída libre.

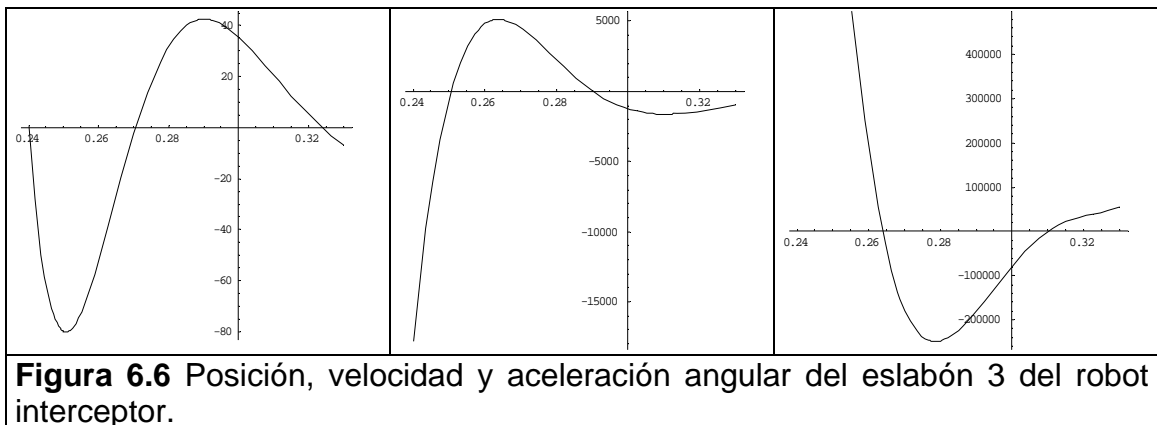


**Figura 6.5** Posición, velocidad y aceleración angular del eslabón 2 del robot interceptor.

$$\text{teta3}(t) = 6028481985 t^6 - 11172648888 t^5 + 8608533131 t^4 - 3528630911 t^3 + 811275646 t^2 - 99160863 t + 5032176 \quad (6.7)$$

$$\text{Vteta3}(t) = 36170891910 t^5 - 55863244440 t^4 + 34434132524 t^3 - 10585892733 t^2 + 1622551292 t - 99160863 \quad (6.8)$$

$$\text{Ateta3}(t) = 180854459550 t^4 - 223452977760 t^3 + 103302397572 t^2 - 21171785466 t + 1622551292 \quad (6.9)$$



**Figura 6.6** Posición, velocidad y aceleración angular del eslabón 3 del robot interceptor.

El efector final tiene una trayectoria de posición con respecto al tiempo que depende de los dos eslabones del robot. En otras palabras, el sistema de ecuaciones que se resolvió incluyó los desplazamientos angulares de los eslabones 1 y 2 y el tercer eslabón dependió de esos datos. Por lo tanto, las aceleraciones que tiene el efector son grandes. El controlador tipo P (proporcional) que tiene implementado solo trabaja la posición y se trata siempre de anticipar la posición a la que debe llegar por un periodo de tiempo previo. Lo anterior, es debido a que no se cuenta con un control de velocidad para este eslabón.

**VI.5 ANÁLISIS DE LA TRAYECTORIA SEGUIDORA TPS**

A continuación se presentan las siguientes gráficas de los componentes X e Y de las funciones de la trayectoria de la partícula seguidora TPS, y las partículas libres: TPL1 y TPL2, para el caso particular del evento de intercepción en estudio.

$$\begin{aligned} X_0(t) &= 0.65 - 1.2169 t \\ Y_0(t) &= 0.75 - 0.442916 t - 4.905 t^2 \end{aligned} \tag{6.10}$$

$$\begin{aligned} X_1(t) &= 0.65 - 0.908978 t \\ Y_1(t) &= 0.80 - 0.160277 t - 4.905 t^2 \end{aligned} \tag{6.11}$$

En las ecuaciones 6.10 y 6.11 se presentan las ecuaciones de las trayectorias que siguen las partículas TPL1 y TPL2, respectivamente. Una vez que se planteó el sistema de ecuaciones generado al aplicar la metodología, se sustituyeron los valores iniciales de la tabla 6.2 y el resultado fueron las ecuaciones mencionadas.

Posición:

$$\begin{aligned} \text{TPSx}(t) &= -174.048 + 2452.81 t - 12875.8 t^2 + 29893.1 t^3 - 25897.1 t^4 \\ \text{TPSy}(t) &= -401.596 + 5604.15 t - 29168.5 t^2 + 67165.8 t^3 - 57748.3 t^4 \end{aligned} \tag{6.12}$$

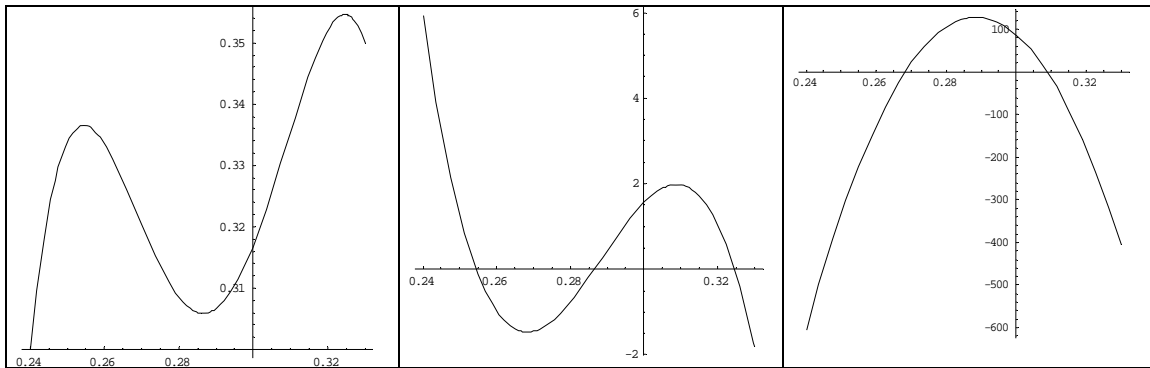
Velocidad:

$$\begin{aligned} \text{VTPSx}(t) &= 2452.81 - 25751.7 t + 89679.4 t^2 - 103589 t^3 \\ \text{VTPSy}(t) &= 5604.15 - 58337 t + 201497 t^2 - 230993 t^3 \end{aligned} \tag{6.13}$$

Aceleración:

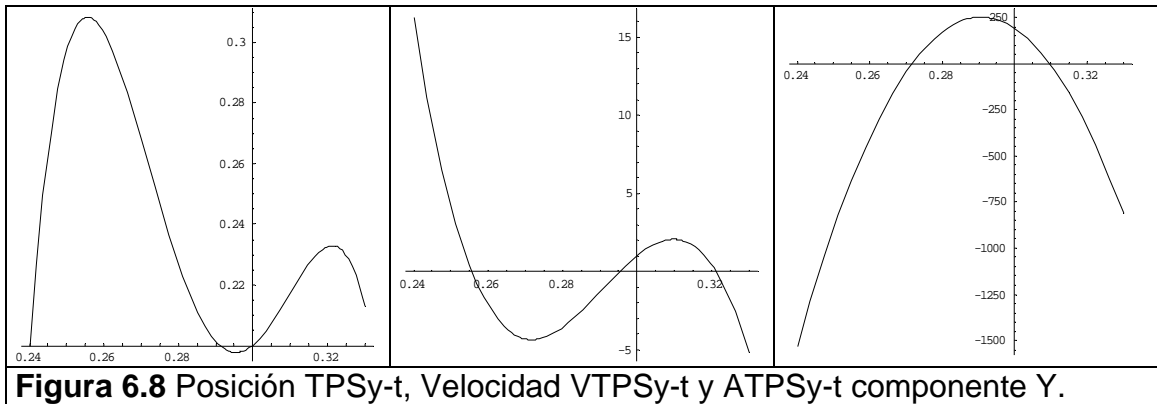
$$\begin{aligned} \text{ATPSx}(t) &= -25751.7 + 179359 t - 310766 t^2 \\ \text{ATPSy}(t) &= -58337 + 402995 t - 692979 t^2 \end{aligned} \tag{6.14}$$

Las ecuaciones 6.12 a la 6.14 contienen un término cuadrático, esto se debe a que la metodología propuesta toma como información las funciones de las trayectorias de las partículas libres TPL1 y TPL2 y por ende la función seguidora toma la misma forma en términos del grado de la función. Para este caso, la función TPS tiene un término cuadrático como máximo grado.



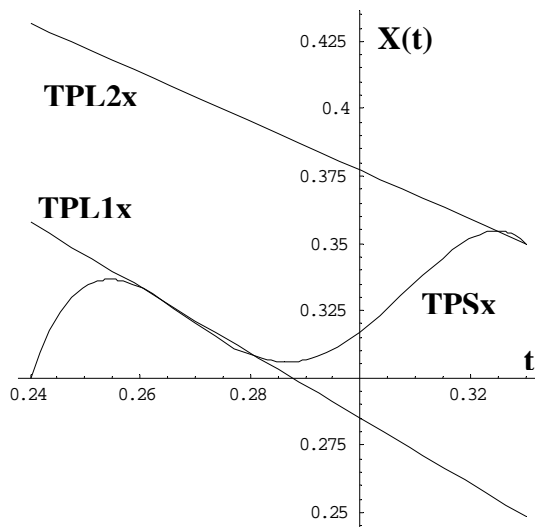
**Figura 6.7** Posición TPSx-t, Velocidad VTPSx-t y ATPSx-t componente X.



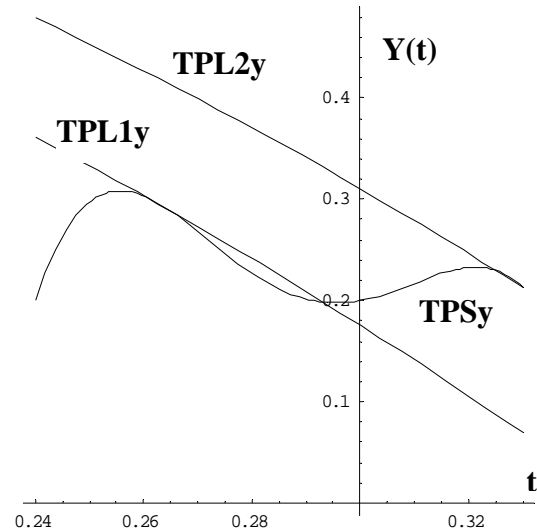


**Figura 6.8** Posición TPSy-t, Velocidad VTPSy-t y ATPSy-t componente Y.

A continuación se presentan las gráficas de la intercepción de las partículas libres TPL1 y TPL2 por medio de la trayectoria TPS, expresada en sus componentes X-Y, ver figura 6.9 y figura 6.10.



**Figura 6.9** Gráfica de la posición-tiempo de la trayectoria TPS (X-t)



**Figura 6.10** Gráfica de la posición-tiempo de la trayectoria TPS (Y-t)

En las figuras 6.9 y 6.10, se puede observar como la trayectoria TPSx toca e iguala las funciones durante un intervalo de tiempo  $\Delta t = 0.2$  s, primero toca a la función TPL1x en el tiempo  $t = 0.26$  y luego toca al componente X de la función TPL2x en el tiempo  $t = 0.34$  s, y permanece igual durante  $\Delta t = 0.2$  s. Lo mismo sucede para el componente Y. Lo anterior representa el fenómeno real donde interviene la posición y el tiempo.

**VI.6 RESULTADOS EXPERIMENTALES DEL CASO PARTÍCULAR EN ESTUDIO**

A continuación se presenta un análisis del evento experimental en estudio. Es evidente que las funciones de velocidad angular de los eslabones del robot interceptor tienen grado cinco, por lo tanto, es conveniente desarrollar las funciones de movimiento con enlaces parabólicos y perfiles de velocidades trapezoidales. A continuación, se presenta la ecuación para calcular la aceleración permitida para cada incremento de velocidad angular. (ver tabla 6.3).

$$\dot{\theta} \geq \frac{4(\theta_f - \theta_0)}{t_f^2} \tag{6.15}$$

**Tabla 6.5** Aceleraciones angulares máximas permitidas (grados)

		Aceleración angular °/s <sup>2</sup>			Posición angular en °		
#	Tiempo s	$\ddot{\theta}_1(t)$	$\ddot{\theta}_2(t)$	$\ddot{\theta}_3(t)$	$\theta_1(t)$	$\theta_2(t)$	$\theta_3(t)$
13	0.24	0	0	0	0	0	0
14	0.26	-74117.1	595956	-521839	-7.41171	59.5956	-52.1839
15	0.28	154735.6	-995256	840521	8.06185	-39.93	31.8682
16	0.3	-114140.6	76279	37861	-3.35221	-32.3021	35.6543
17	0.32	-22765.2	326927.95	-304162.7	-5.62873	0.390695	5.23803
18	0.34	27911.8	155049.05	-182960.3	-2.83755	15.8956	-13.058
19	0.36	-105065.5	-453851	558916	-13.3441	-29.4895	42.8336

$$\dot{\theta} = 2 \frac{\Delta\theta}{\Delta t} \tag{6.16}$$

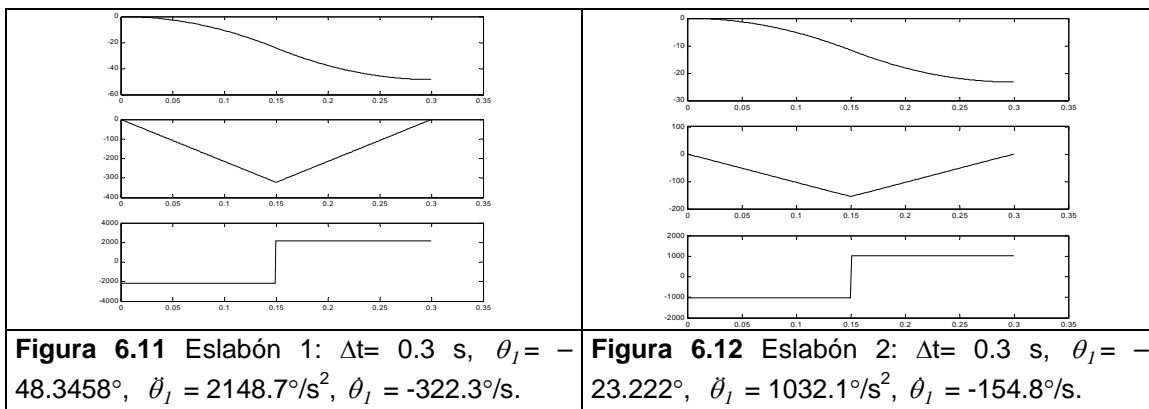
**Tabla 6.6** Velocidades angulares calculadas (grados/s)

#	Tiempo s	$\dot{\theta}_1(t)$	$\dot{\theta}_2(t)$	$\dot{\theta}_3(t)$
13	0.24	0	0	0
14	0.26	-741.171	3546.82934	-1010.68911
15	0.28	1547.356	8178.6178	-2718.68096
16	0.3	-1141.406	-977.747869	-109.210223
17	0.32	-227.652	3863.48322	5152.24358
18	0.34	279.118	1021.40349	-2010.55275
19	0.36	-1050.655	4992.86029	-1555.22288

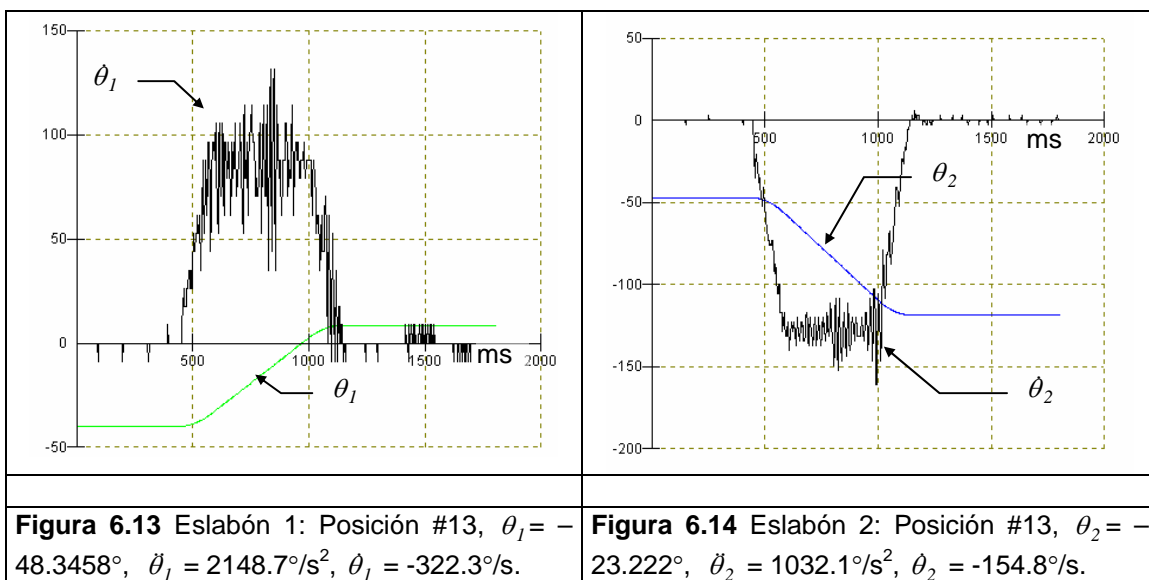
En la tabla 6.6 se presentan las aceleraciones de las articulaciones del robot interceptor con un incremento del 1% sobre su valor nominal, esto con el fin de alcanzar la velocidad máxima angular de los eslabones. Lo anterior permitirá alcanzar la máxima velocidad disminuyendo al máximo la aceleración, ver figura 6.11, que en este caso es importante debido a que es directamente proporcional a la fuerza, según la segunda ley de Newton, evitando así forzar al controlador y al sistema con impactos debido a las altas aceleraciones. Cabe señalar que lo anterior es la interpretación física.

Tabla 6.7 Aceleraciones con +1% de tolerancia ( $^{\circ}/s^2$ )				
#	Tiempo s	$\ddot{\theta}_1(t)$	$\ddot{\theta}_2(t)$	$\ddot{\theta}_3(t)$
13	0.24	0.0	0.0	0.0
14	0.26	-74858.3	601915.6	-527057.4
15	0.28	156283.0	-1005208.6	848926.2
16	0.3	-115282.0	77041.8	38239.6
17	0.32	-22992.9	330197.2	-307204.3
18	0.34	28190.9	156599.5	-184789.9
19	0.36	-106116.2	-458389.5	564505.2

En la figura 6.11 se presenta el tipo de movimiento que tendrán los eslabones del robot interceptor, que será del tipo parabólico con perfil de velocidad trapezoidal y aceleración constante. Para el caso en que el eslabón 1, éste rota desde la posición # 13 a la posición #14, es decir, de  $0^{\circ}$  a  $-7.41171^{\circ}$  en un intervalo de tiempo de 0.02 s. y con una aceleración de  $-74858.3^{\circ}/s^2$  ver tabla 6.7 y tabla 6.6.



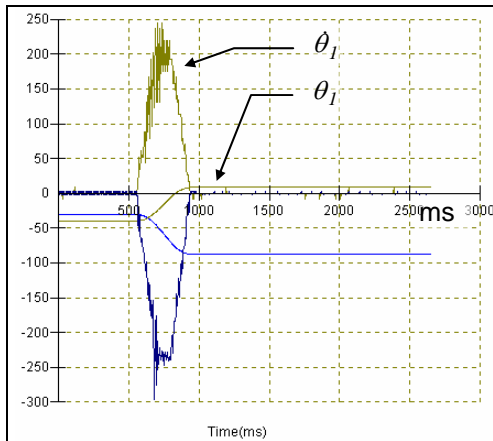
A continuación se presentan los resultados experimentales del evento de intercepción. Para alcanzar la posición de inicio de intercepción en estudio, ver tabla 6.7.



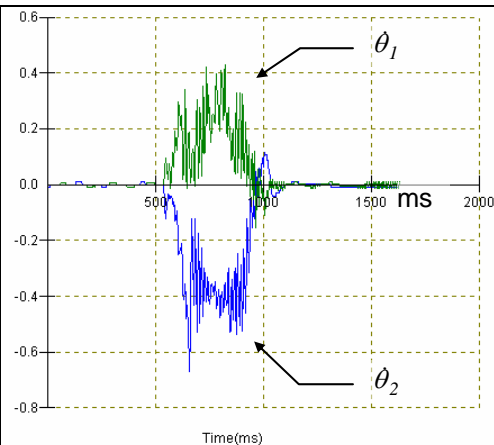
Es importante mencionar que el signo de rotación cambia en el movimiento del eslabón 1 del robot real, y el desplazamiento se realizó desde la posición de home, y que la velocidad angular sufre los efectos debido a la vibración del sistema. Las ganancias que presentaron un mejor desempeño se presentan en la tabla 6.8:

**Tabla 6.8** Ganancias posición # 13.

	Kp	Kd	Ki	Kvel	Kvelff
Eslabón1	5.0	2.0	0.8	0.0	0.0
Eslabón2	5.0	2.0	0.8	0.0	0.0

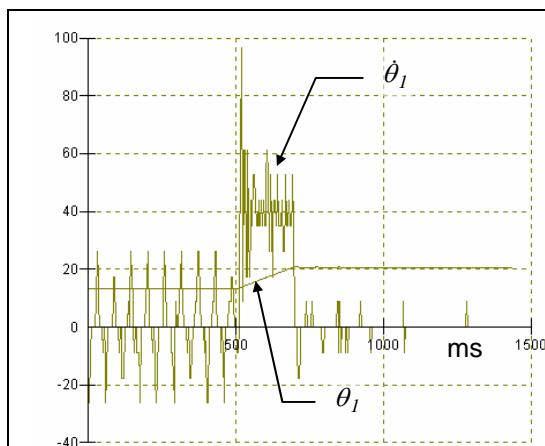


**Figura 6.15** Posición #13, gráficas combinadas de posición y velocidad angular

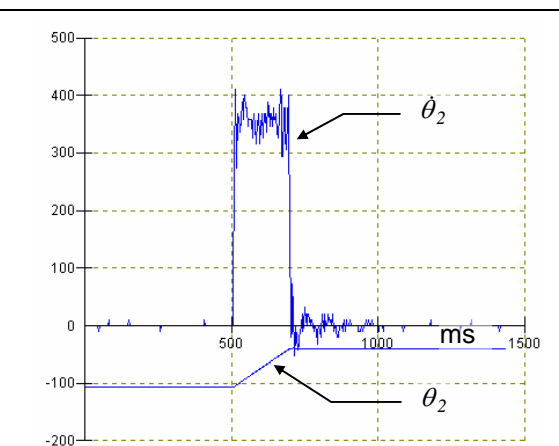


**Figura 6.16** Posición #13, errores de seguimiento de velocidad

Las velocidades alcanzadas, ver tabla 6.6 y figuras 6.17 a la 6.38, son velocidades sobre la trayectoria interpolada, es decir, debido a que el movimiento es interpolado la velocidad de cada eslabón se combina para generar una velocidad sobre la trayectoria interpolada que en este caso es un movimiento rectilíneo.

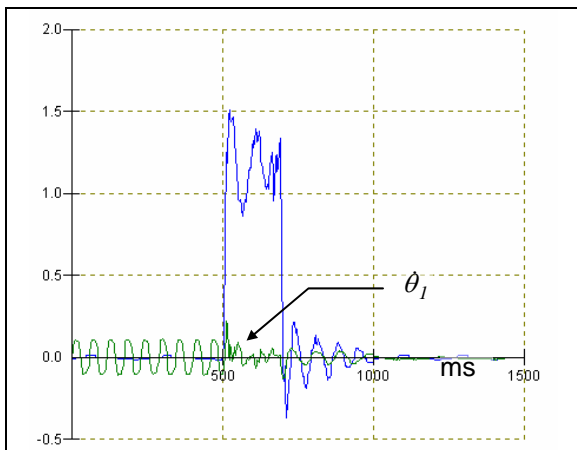


**Figura 6.17** Eslabón 1: Posición #14,  $\theta_1 = -7.4^\circ$ ,  $\ddot{\theta}_1 = 74858.3^\circ/s^2$ ,  $\dot{\theta}_1 = -741.2^\circ/s$ .



**Figura 6.18** Eslabón 2: Posición #14,  $\theta_2 = 59.6^\circ$ ,  $\ddot{\theta}_2 = 601915.6^\circ/s^2$ ,  $\dot{\theta}_2 = 3546.8^\circ/s$ .

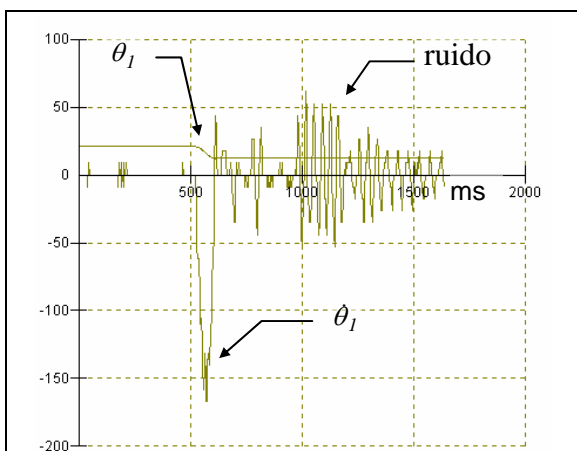
Nota: Escala 10:1



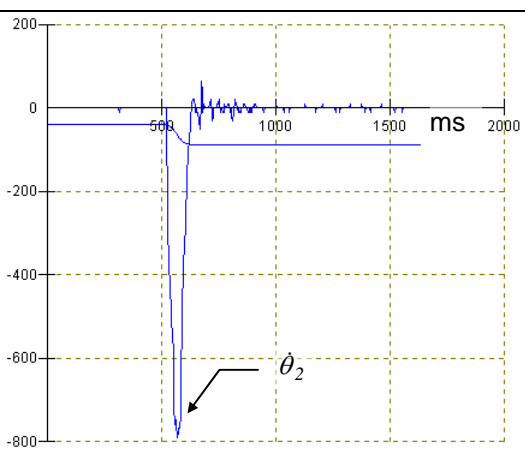
**Tabla 6.9** Ganancias posición # 14.

	Kp	Kd	Ki	Kvel	Kvelff
Eslabón1	10.0	5.0	1.5	5.0	1.0
Eslabón2	7.0	3.0	1.0	5.0	1.0

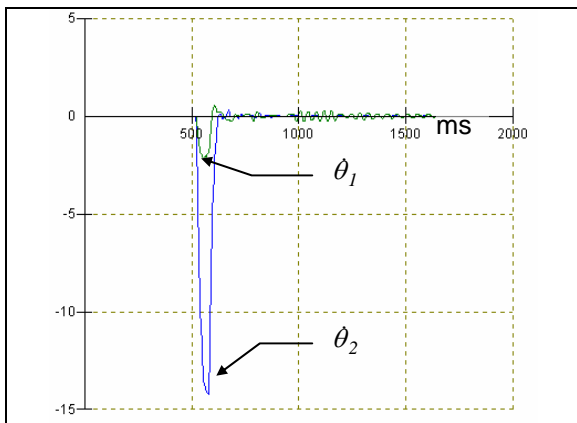
**Figura 6.19** Errores de seguimiento de velocidad



**Figura 6.20** Eslabón 1: Posición #15,  $\theta_1 = -8.1^\circ$ ,  $\ddot{\theta}_1 = 156283.0^\circ/s^2$ ,  $\dot{\theta}_1 = -1547.4^\circ/s$ .



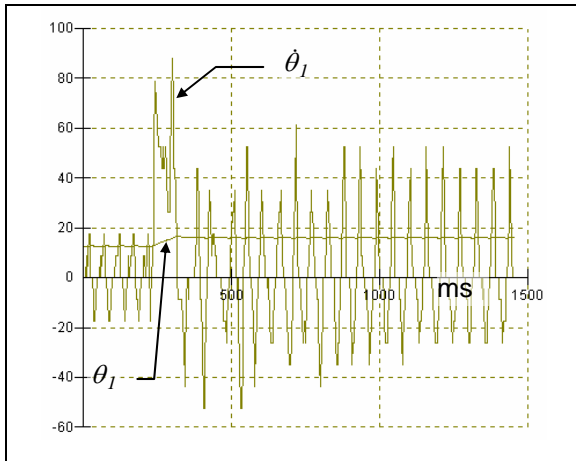
**Figura 6.21** Eslabón 2: Posición #15,  $\theta_2 = -39.9^\circ$ ,  $\ddot{\theta}_2 = -1005208.6^\circ/s^2$ ,  $\dot{\theta}_2 = 8178.6^\circ/s$ .



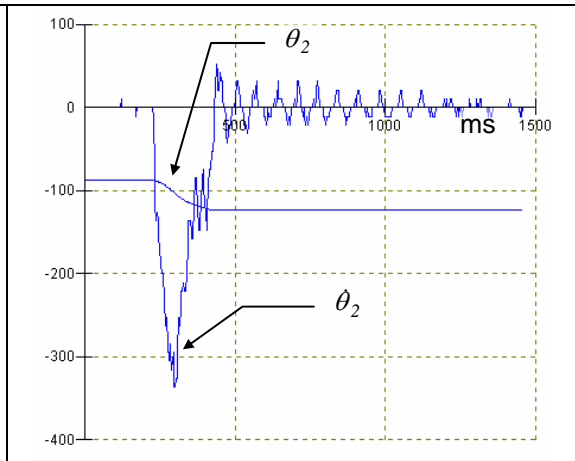
**Tabla 6.10** Ganancias posición # 15.

	Kp	Kd	Ki	Kvel	Kvelff
Eslabón1	1.0	0.5	1.0	1.5	1.0
Eslabón2	1.2	0.7	2.0	3.1	2.0

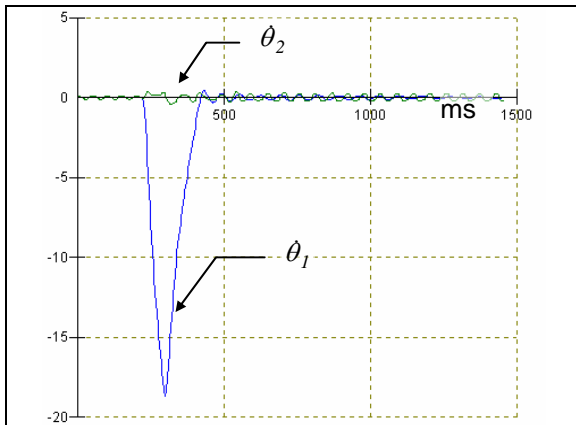
**Figura 6.22** Errores de seguimiento de velocidad



**Figura 6.23** Eslabón 1: Posición #16,  $\theta_1 = -3.4^\circ$ ,  $\ddot{\theta}_1 = -115282.0^\circ/s^2$ ,  $\dot{\theta}_1 = -1141.4^\circ/s$ .



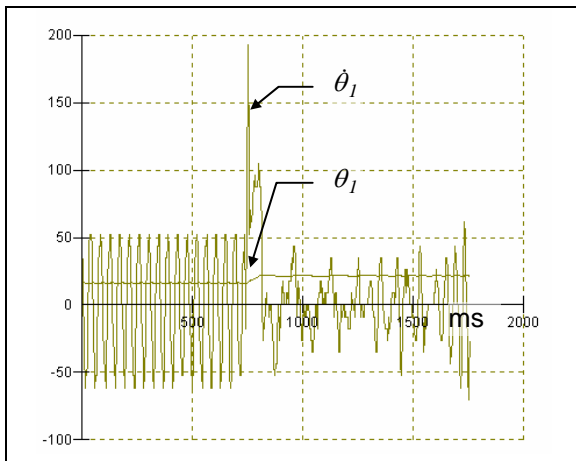
**Figura 6.24** Eslabón 2: Posición #16,  $\theta_2 = -32.3^\circ$ ,  $\ddot{\theta}_2 = -77041.8^\circ/s^2$ ,  $\dot{\theta}_2 = -977.7^\circ/s$ .



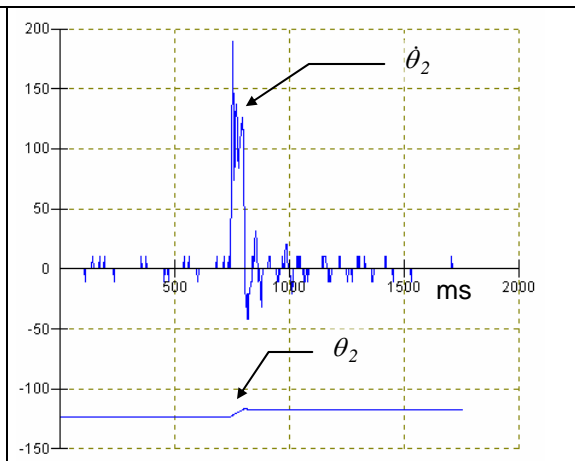
**Figura 6.25** Errores de seguimiento de velocidad

**Tabla 6.11** Ganancias posición # 16.

	Kp	Kd	Ki	Kvel	Kvelff
Eslabón1	0.3	1.2	0.8	1.0	1.0
Eslabón2	0.3	1.2	0.8	1.0	1.0

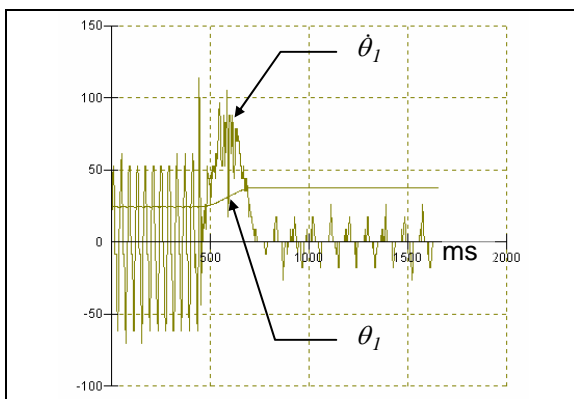
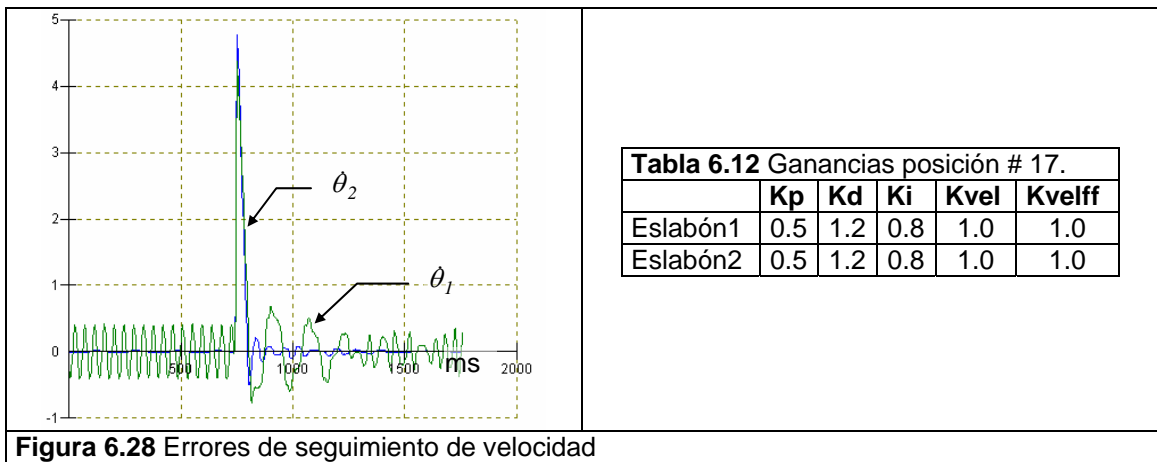


**Figura 6.26** Eslabón 1: Posición #17,  $\theta_1 = -5.6^\circ$ ,  $\ddot{\theta}_1 = -22992.9^\circ/s^2$ ,  $\dot{\theta}_1 = -227.7^\circ/s$ .

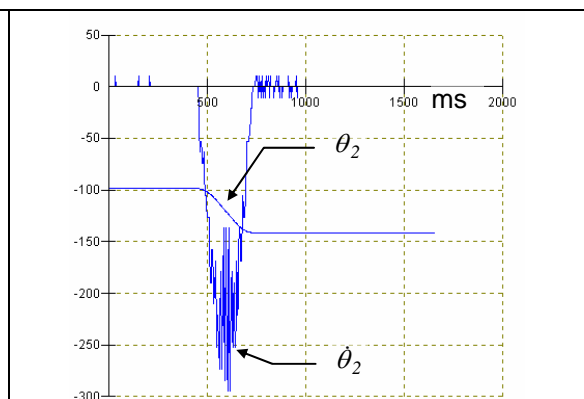


**Figura 6.27** Eslabón 2: Posición #17,  $\theta_2 = 0.4^\circ$ ,  $\ddot{\theta}_2 = 330197.2^\circ/s^2$ ,  $\dot{\theta}_2 = 3863.5^\circ/s$ .

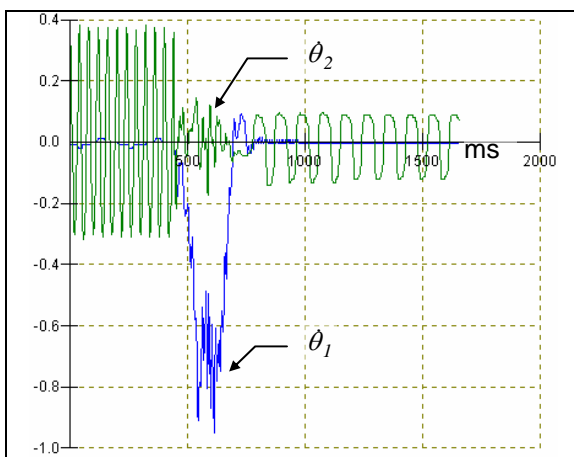
Escala 2:1



**Figura 6.29** Eslabón 1: Posición #18,  $\theta_1 = -2.8^\circ$ ,  $\ddot{\theta}_1 = -28190.9^\circ/s^2$ ,  $\dot{\theta}_1 = -279.1^\circ/s$ .

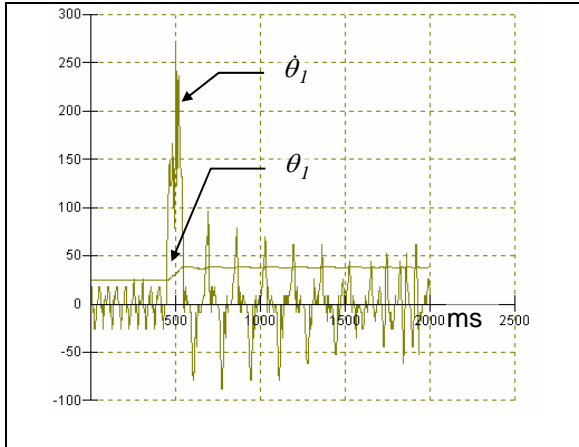


**Figura 6.30** Eslabón 2: Posición #18,  $\theta_2 = 15.9^\circ$ ,  $\ddot{\theta}_2 = 156599.5^\circ/s^2$ ,  $\dot{\theta}_2 = 1021.4^\circ/s$ .

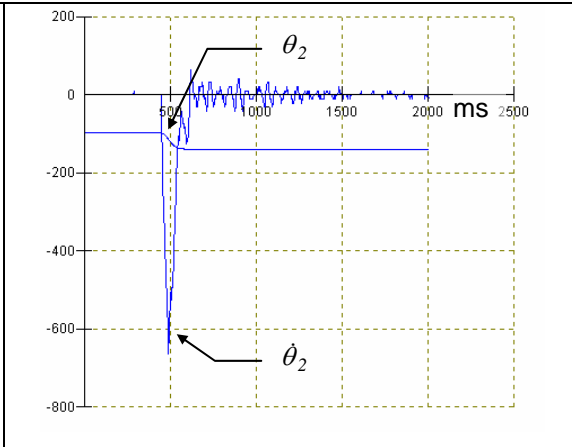


**Figura 6.31** Errores de seguimiento de velocidad  
*Escala 2:1*

	Kp	Kd	Ki	Kvel	Kvelff
Eslabón1	1.0	0.8	0.8	1.0	1.0
Eslabón2	1.0	0.8	0.8	1.0	1.0

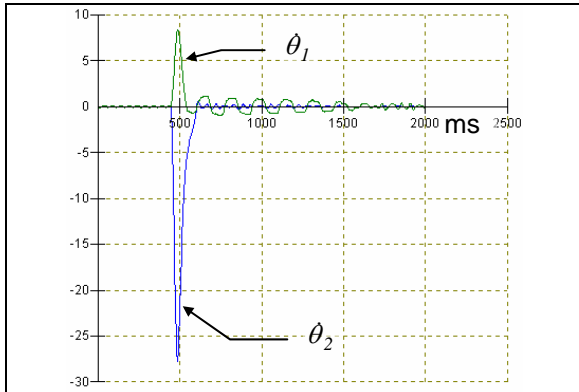


**Figura 6.32** Eslabón 1: Posición #19,  $\theta_1 = -13.3^\circ$ ,  $\ddot{\theta}_1 = -106116.2^\circ/s^2$ ,  $\dot{\theta}_1 = -1050.655^\circ/s$ .



**Figura 6.33** Eslabón 2: Posición #19,  $\theta_2 = -29.5^\circ$ ,  $\ddot{\theta}_2 = -458389.5^\circ/s^2$ ,  $\dot{\theta}_2 = 4992.9^\circ/s$ .

Escala 4:1

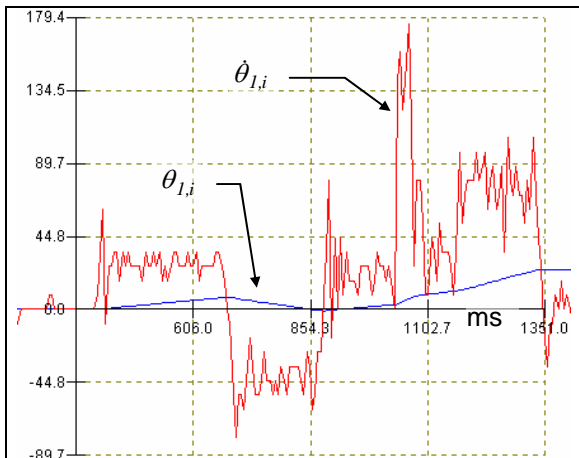


**Figura 6.34** Errores de seguimiento de velocidad

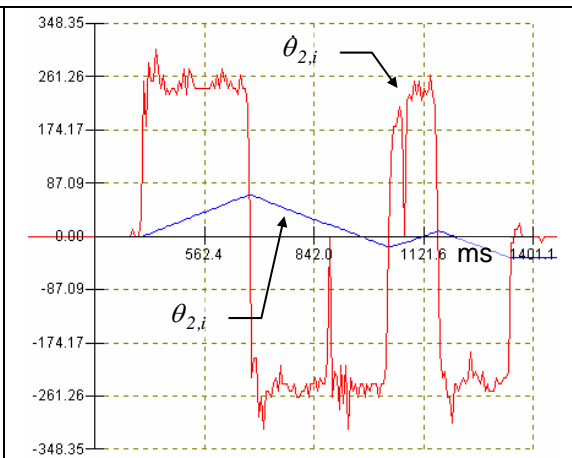
**Tabla 6.14** Ganancias posición # 19.

	Kp	Kd	Ki	Kvel	Kvelff
Eslabón1	0.5	0.8	0.8	1.0	1.0
Eslabón2	0.5	0.8	0.8	1.0	1.0

A continuación se presentan las gráficas de la secuencia de movimiento completa desde la posición #13 a la posición #19. Cabe mencionar que se tomaron los promedios de velocidad y aceleración, ver figura 6.35 a la 6.37.



**Figura 6.35** Historial de la secuencia de movimiento  $i=1,6$  del eslabón 1.



**Figura 6.36** Historial de la secuencia de movimiento  $i=1,6$  del eslabón 2.

Escala 15:1



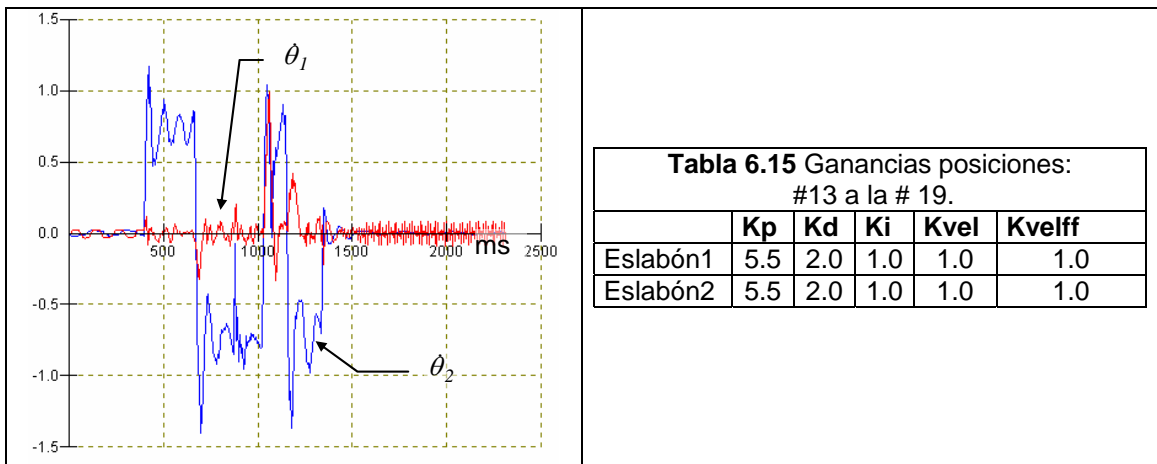


Figura 6.37 Errores de seguimiento de velocidad de la secuencia.

Escala 15:1

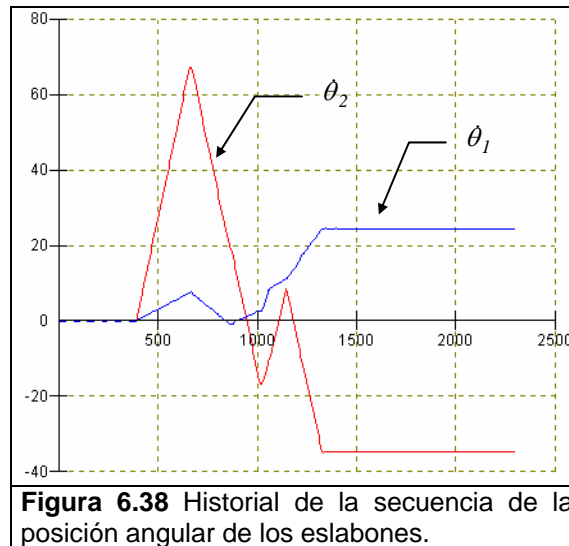


Figura 6.38 Historial de la secuencia de la posición angular de los eslabones.

En las figuras 6.13 a la 6.37 se muestra el comportamiento real del robot de cada uno de sus eslabones en el evento de intercepción en estudio.

### VI.7 EL GRIPPER DEL ROBOT INTERCEPTOR

El gripper del robot interceptor tiene un control proporcional que ajusta la posición angular con respecto a los eslabones anteriores. La relación de voltaje-ángulo está dada por la figura 6.39, la cual muestra la línea de tendencia, ecuación (6.17).

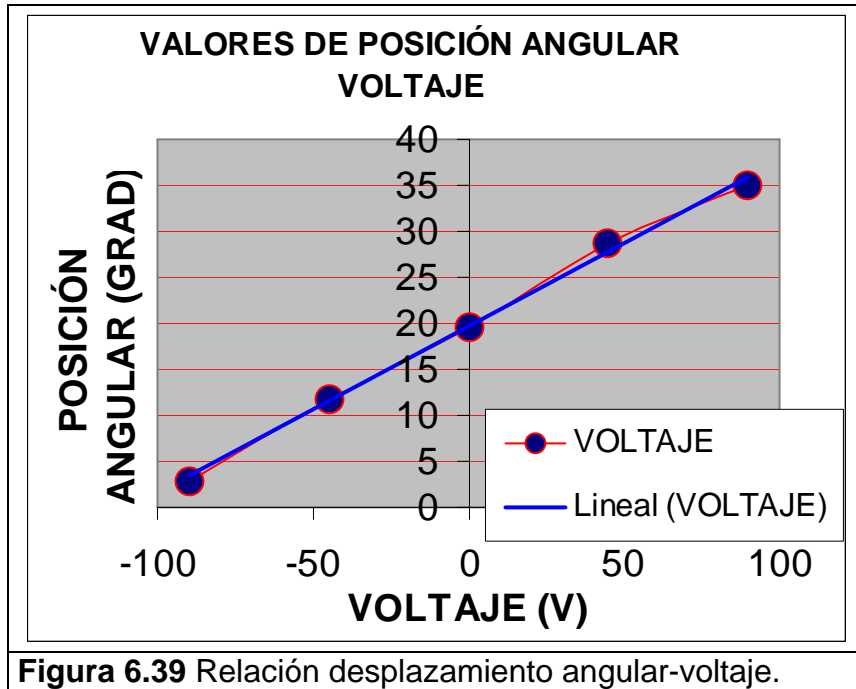


Figura 6.39 Relación desplazamiento angular-voltaje.

$$V = 0.1801\theta + 19.588 \tag{6.17}$$

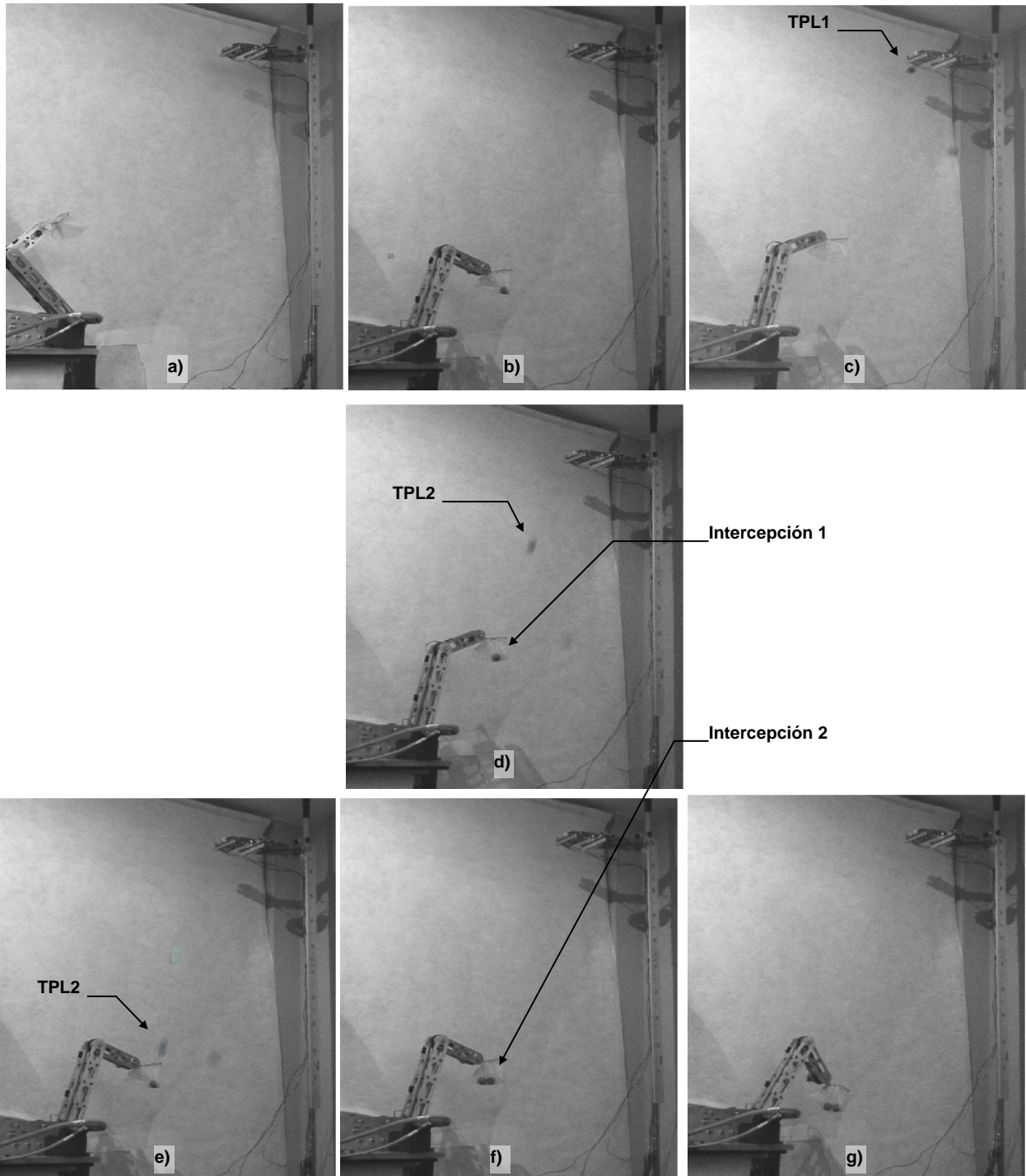
los resultados fueron los siguientes, ver tabla 6.16.

#	Tiempo	Voltaje (V)	$\theta_3(t)$ (°)
13	0.24	23.2	0
14	0.26	15.9	-52.1839
15	0.28	13.5	31.8682
16	0.3	18.4	35.6543
17	0.32	21.1	5.23803
18	0.34	24.9	-13.058
19	0.36	31.2	42.8336

### VI.8 ANÁLISIS DE RESULTADOS DEL CASO PARTICULAR DE TIRO PARABÓLICO DEL EVENTO EN ESTUDIO

Como se puede observar en las gráficas anteriores el movimiento de los eslabones del robot se encuentra afectado fuertemente por la vibración que ocasiona el tren de engranaje. La función del tren de engranaje es reducir la respuesta en velocidad de los motores, ya que ésta es muy alta y en desplazamientos angulares pequeños el motor presenta un desempeño pobre. El tren de engranaje fue implementado para evitar el problema de alta velocidad, sin embargo, esto en la práctica ocasionó una vibración alta. El perfil de velocidad trapezoidal alcanza la referencia con mucha oscilación, sin embargo, cumple con la posición con un error aceptable.

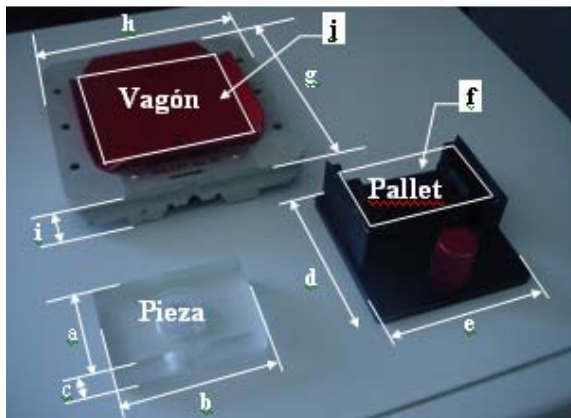
En la figura 6.4 se presenta la secuencia de imágenes del evento en estudio, se realizaron varios intentos para interceptar las dos esferas con la misma secuencia de movimiento obteniendo un promedio de 6 intercepciones por cada 10 intentos, los fracasos se debieron principalmente al desajuste del tren de engranaje y aflojamiento de las bandas de los eslabones, ya que el movimiento de los motores era demasiado rápido y el frenado resultaba en un impacto sobre las poleas.



**Figura 6.40** Secuencia de imágenes del evento particular experimental de intercepción

**VI.9 PARTE EXPERIMENTAL CON LA ESTACION SISTEMA DE MANUFACTURA FLEXIBLE (FMS).**

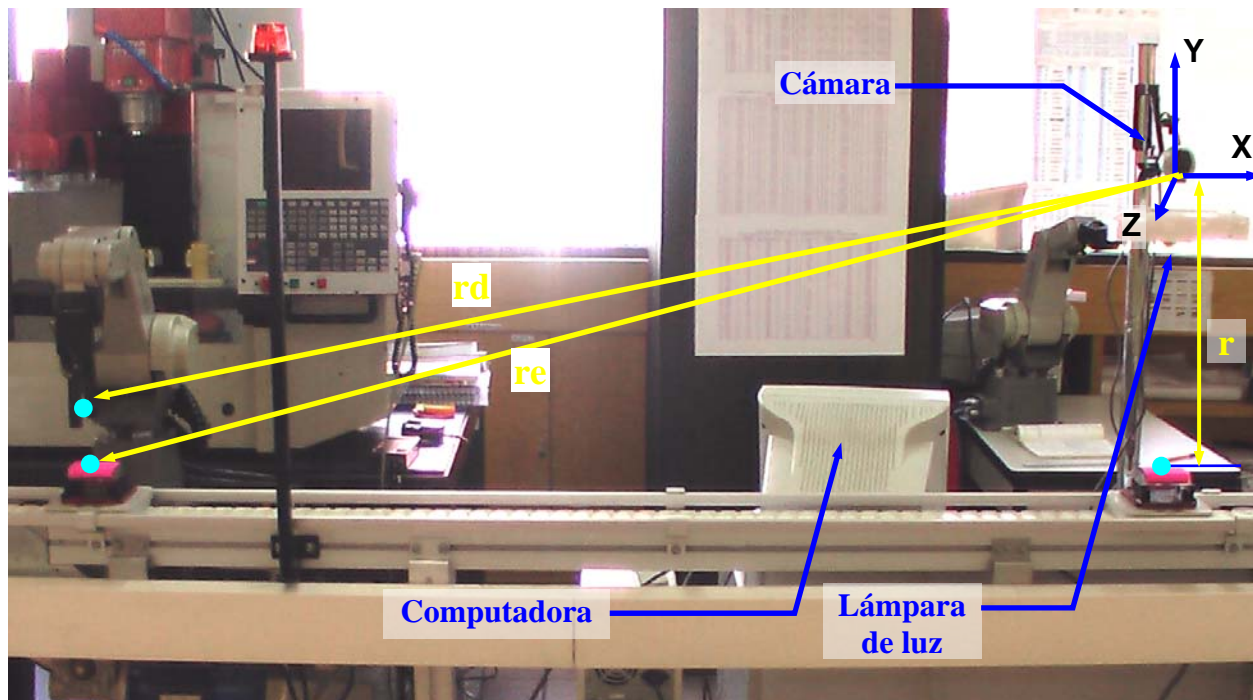
En este punto se presenta el desarrollo y resultados del segundo experimento utilizando en este caso el robot industrial que se encuentra en el Sistema de Manufactura Flexible (FMS). Las dimensiones de las piezas, el pallet y el vagón que corren sobre la banda transportadora están en la tabla 6.17, ver figura 6.41.



**Tabla 6.17**  
Dimensiones del vagón, pallet y pieza en metros

Pieza	Pallet	Vagón
a=0.05	d= 10.00	g=0.15
b=0.02	e= 10.00	h=0.15
c=0.08	f= 0.0004 m <sup>2</sup>	l=0.025
		j=0.025 m <sup>2</sup>

**Figura 6.41** Dimensiones de la pieza, vagón y pallet.



**Figura 6.42a** Dimensiones del sistema integrado visión-robot Mitsubishi®.

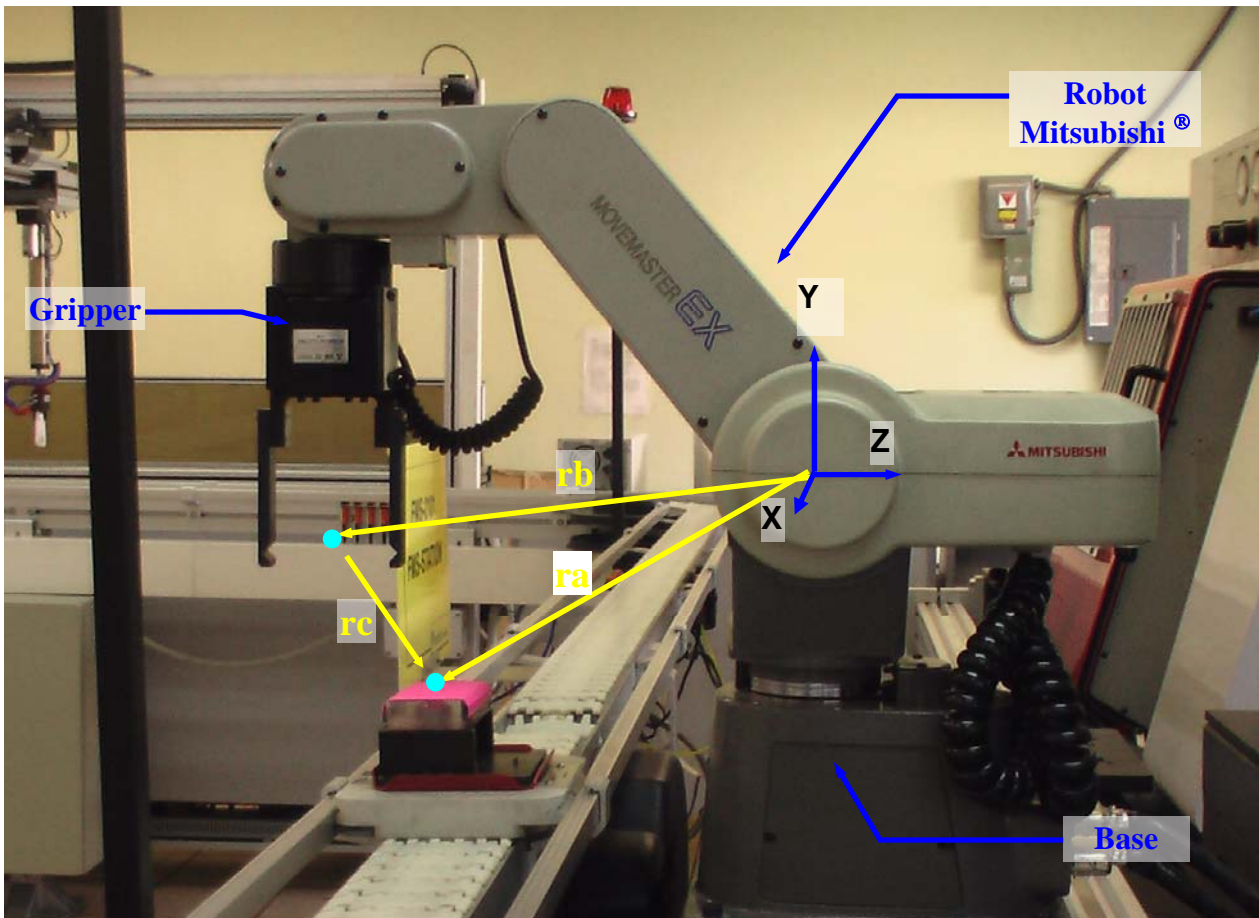


Figura 6.42b Dimensiones del robot Mitsubishi®.

En la figura 6.42a y 6.42b se presentan las distancias necesarias que tendrá el robot y el sistema de visión para realizar la parte experimental. Las dimensiones se establecen a través de variables con el fin de hacerlo más flexible a la hora de realizar ajustes, ver tabla 6.18.

<b>Tabla 6.18</b>	
Vectores que definen las posiciones de los componentes del sistema	
<b>coordenadas</b>	<b>Unidad</b>
$r = (0, -0.47, 0)$	m
$ra = (0, -0.155, -0.25)$	m
$rb = (0, 0.06, -0.32)$	m
$rc = ra - rb = (0, -0.95, 0.07)$	m
$rd = (-1.785, -0.37, 0)$	m
$re = (-1.8, -0.47, 0)$	m

A continuación, se muestran las posiciones máximas y mínimas de los centroides de las piezas rectangulares que el sistema de visión puede detectar, que es el alcance visual máximo dentro de su campo de visión. Los pallets deben tener una distancia relativa entre ellos mínima de 0.15 m y una distancia máxima entre ellos de 0.45 m para que el sistema de visión pueda “verlos”. Con la información obtenida se calculará la velocidad

y la distancia que separa las piezas para generar la trayectoria TPS que el robot debe seguir para atraparlas dentro del área de trabajo del robot.

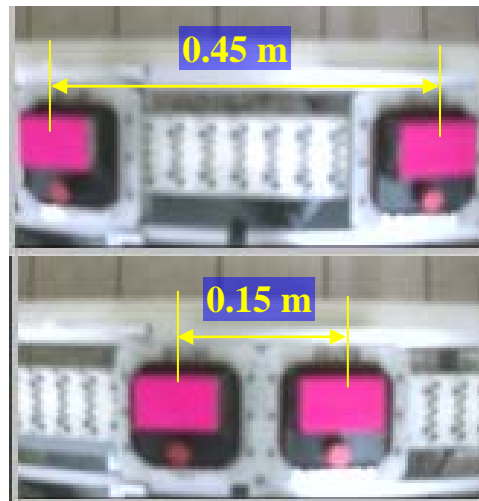
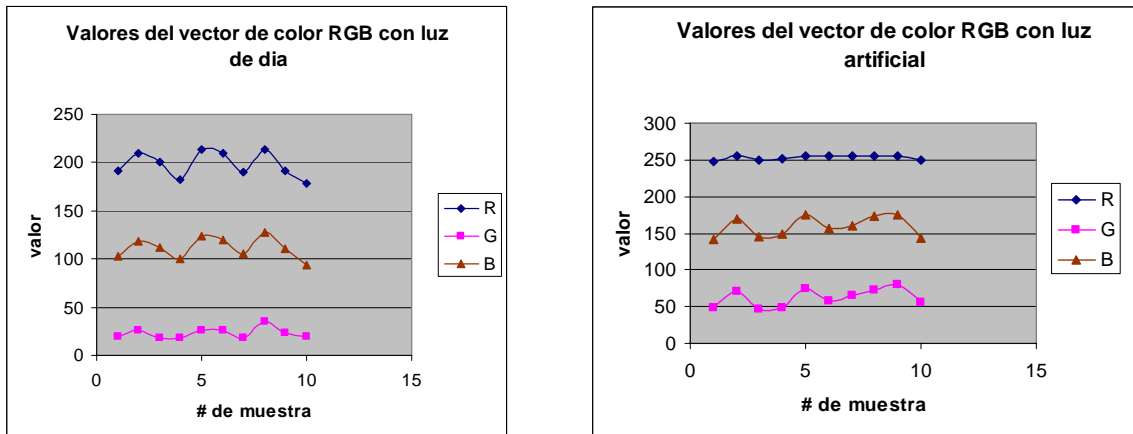


Figura 6.43 Distancia (d) máxima y mínima

#### VI.10 SECUENCIA DE PASOS DESCRIPTIVOS DE LA PARTE EXPERIMENTAL

A continuación se presenta la secuencia de pasos a seguir para que el robot Mitsubishi® capture las dos piezas sobre la banda transportadora en movimiento:

- 1.- Se abre el puerto serial de comunicaciones de la computadora y se establece comunicación con el controlador del Robot Mitsubishi®, protocolo RS-232.
- 2.- Se envía el comando “NT” para que el robot se mueva a la posición de “Home” y se leen las coordenadas desde el buffer, las coordenadas son: (Hombro, brazo, antebrazo, pitch angle, roll angle)=(+51.0,-88.3,+737.1,+10.9,+179.9) grados
- 3.- Se inicializa y se configura el sistema de visión para que capture 30 cuadros/seg. y capture las imágenes en el formato RGB de 24 bits de profundidad de color.
- 4.- Se fija la resolución de la imagen a 120x160 píxeles.
- 5.- Se detecta y graba el color de la pieza para que el sistema de visión la reconozca y la discrimine por el color: se utilizó el color rosa cuyo vector de color RGB es (Red-Green,Blue) = (231,47,139) y un umbral de tolerancia de  $\pm 48$ , ver figura 6.44. En esta etapa de la secuencia se presenta el problema del cambio de intensidad de luz en el medio ambiente, en la figura 6.44 se puede observar el fenómeno, las mediciones fueron hechas en un intervalo de tiempo de 20 minutos.



**Figura 6.44** Variación en las intensidad de luz del medio ambiente sobre la pieza de la figura 6.43.

Por lo anterior, se decidió cambiar del espacio RGB al espacio HSV [6.1]. Lo anterior permite segmentar el color de la pieza para separarla del resto de la imagen, recordando que la segmentación es un proceso que divide o parte una imagen en regiones, logrando de esta forma aislar cierta región de interés. La idea principal es segmentar el color rosa de la imagen en la figura 6.45.



**Figura 6.45** Color de la pieza a analizar

Para realizar la segmentación de imágenes en color existen principalmente dos espacios en los cuales se puede trabajar, estos son:

- Segmentación en el espacio de color HSI
- Segmentación en el espacio vectorial RGB

Cuando se desea segmentar una imagen con base en el color, y además se requiere trabajar con los planos individuales de dicha imagen, lo más conveniente es trabajar en el espacio HSI (hue, saturation, intensity). Aquí, se utilizará un modelo muy parecido al HSI llamado HSV (hue, saturation value) [6.2]. El significado de las componentes son:

- Hue, el color es representado por una forma más conveniente en el plano hue, ya que este plano es el que contiene la información acerca del color de la imagen.

- Saturation, este plano es utilizado como una máscara para aislar regiones de interés en el plano de color (hue).
- Value, este plano es muy poco usado para la segmentación de imágenes en color, porque no contiene información del color.

La imagen que se adquirió desde de la cámara se encuentra en el espacio RGB, para convertirla al espacio HSI es necesario aplicar las siguientes ecuaciones [6.2] a cada una de las componentes RGB. La componente de cada color está dada por ecuaciones (6.18) a la ecuación (6.20).

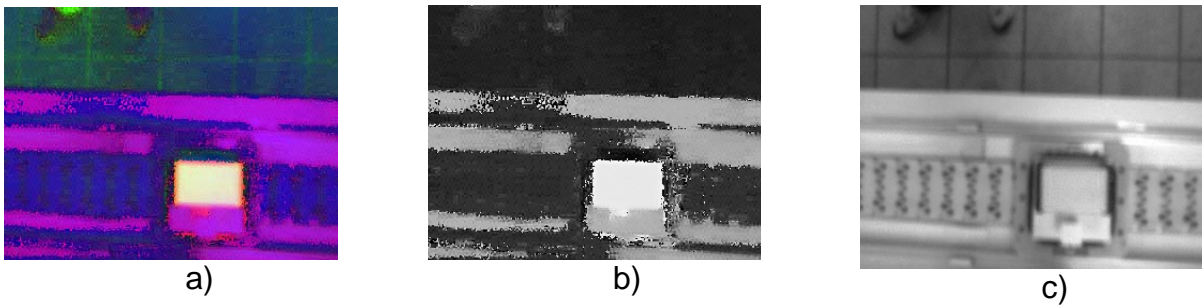
$$H = \begin{cases} \theta & \text{si } B \leq G \\ 360 - \theta & \text{si } B > G \end{cases} \quad (6.18)$$

$$\theta = \cos^{-1} \left\{ \frac{1/2[(R-G) + (R-B)]}{[(R-G)^2 + (R-B)(G-B)]^{1/2}} \right\} \quad (6.19)$$

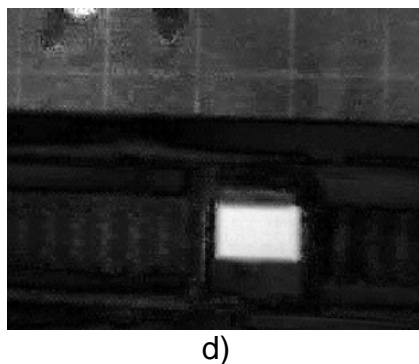
La componente de saturación esta dada por:

$$S = 1 - \frac{3}{(R+G+B)} [\min(R,G,B)] \quad (6.20)$$

A continuación se muestran los resultados obtenidos del proceso de segmentación:



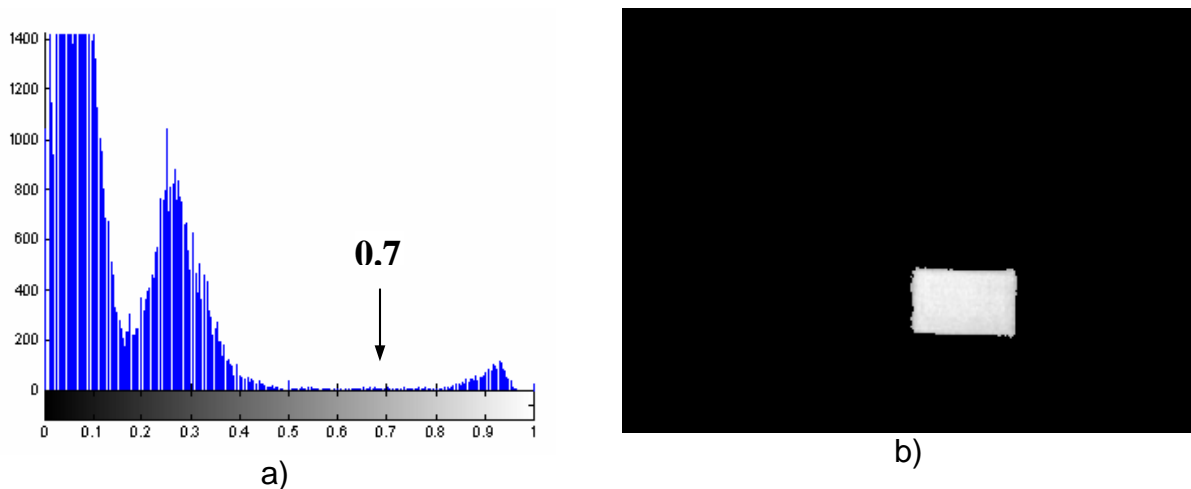
**Figura 6.46** Imagen en el espacio HSV: a) imagen original, b) Componente Hue, c) Componente Value.



**Figura 6.46** d) Componente saturación, (s).

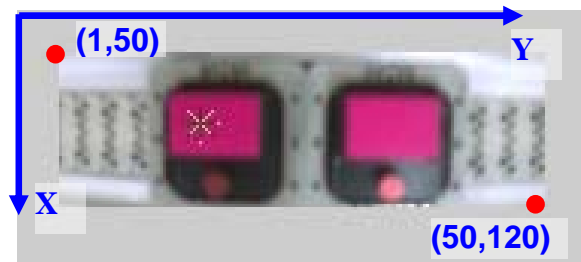
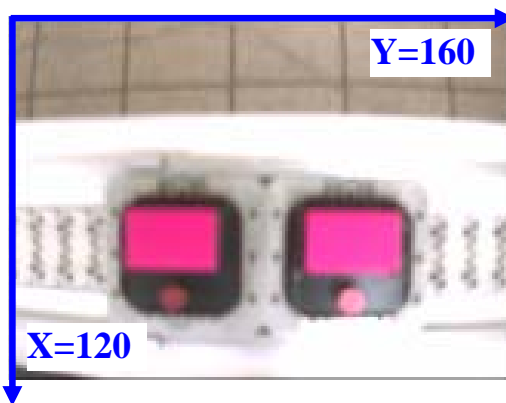


Se puede observar en la figura 6.46d la componente de saturación de la imagen en HSV es la que mejor segmenta la pieza, por lo tanto, se utilizará la componente **s**. A continuación se presenta una segmentación por histograma para la componente **s**.



**Figura 6.47** a) Histograma del componente **s** , b) Resultado de la binarización

Como se puede observar, las componentes más significativas de la imagen se encuentran por encima del valor de 0.7, es por eso que se eligió dicho umbral para segmentar. El resultado obtenido de la segmentación de la pieza por color se muestra en la figura 6.47b.



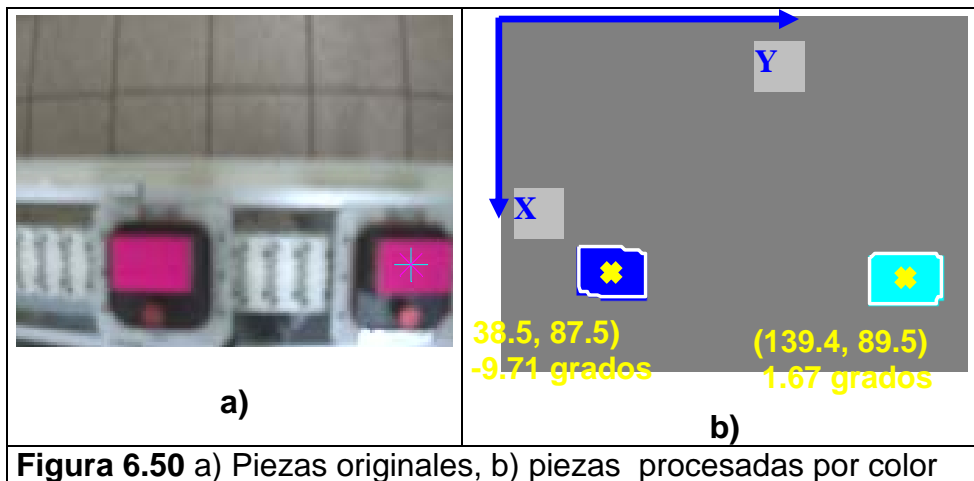
**Figura 6.48** Vector de color **Figura 6.49** Recorte de la imagen [1 50  
RGB=(231,47,139) para el rosa mostrado 160 50].

6.- Para lograr menor tiempo de cálculo en el procesamiento de la imagen se recorta la imagen alrededor de la región de interés, que en este caso es: [Xmin Ymin ancho alto] = [1 50 160 50], ver figura 6.49.

7.- Detección de las piezas por el sistema de visión. En esta tapa, el sistema de visión se encuentra censando cada 2 ms, si en la imagen capturada se encuentra el color de la pieza que coincide con el color almacenado en la memoria en código de color RGB, que ha sido grabado previamente, el sistema se dispara y empieza a procesar la información visual.

8.- Al identificar el color predefinido, el sistema de visión toma varias fotos con tiempos determinados y realiza la conversión a escala de grises de las imágenes, a la cuales, se les calcula el centroide, la posición y la velocidad. El color se codifica en RGB (Red,Green,Blue) que será el vector de color, para este caso el color rosa mostrado en la figura 6.50.

9.- Finalmente, con la velocidad y distancia entre las piezas obtenidas por el sistema de visión, se calcula la trayectoria TPS que debe seguir el robot para interceptar las piezas y capturarlas.



### VI.11 HISTOGRAMAS DE VELOCIDAD Y DEL VECTOR DE COLOR RGB.

Para considerar las posibles variaciones debidas a los cambios de luz que se producen en el medio ambiente, se realizaron 35 eventos para calcular los histogramas que se presentan en la figura 6.51. Estos, se obtuvieron para determinar la velocidad de la banda y estimar las variaciones de velocidad. Los eventos se realizaron en intervalo de tiempo de 10:05 a 12:11 de la mañana, las condiciones climáticas fueron de un día nublado con lluvia y el color de la pieza era rosa, ver figura 6.51b.

Lo anterior permitió establecer la eficiencia del sistema de visión. Es importante mencionar que durante los 35 eventos, hubo 2 fallas debidas al programa, inherentes al CPU. Ocasionando una fuerte variación en los valores obtenidos. Se puede deducir de la figura 6.51a que la velocidad de la banda promedio es de 0.17 m/s aprox. Y que la distancia mínima de las piezas es de 51.5 píxeles. El vector de color varia como se muestra en la figura 6.51b y se establece entonces que el promedio es de  $RGB=(180,55,105)$ .

En la figura 6.52 se muestra la interfase gráfica del sistema de visión desarrollada bajo el lenguaje de programación Matlab<sup>®</sup> 7.0, debido a que cuenta con la capacidad para enviar y recibir información desde el puerto serial de la computadora a el controlador del robot industrial, y adquiere la información visual desde el puerto USB donde se conecta la cámara. Este sistema de visión puede calcular la velocidad y la distancia a la que se encuentran las piezas en movimiento. La discriminación es por el color grabado previamente de la pieza. Aunque, pasen piezas de otro color, el único color que detecta

es el grabado. Después de adquirir los datos, estos se utilizan para realizar los cálculos relacionados con la obtención de las variables articulares del robot industrial y la velocidad angular a la cual deben rotar cada uno de los eslabones para seguir la trayectoria TPS y capturar las piezas. El sistema de visión además, puede detectar dos o más piezas y realizar el procesamiento de las imágenes para calcular el área, el centroide y la orientación de las mismas. El ordenador que se utilizó tiene un procesador pentium IV® con 500 Mb de RAM y un reloj de 1.7 Ghz.

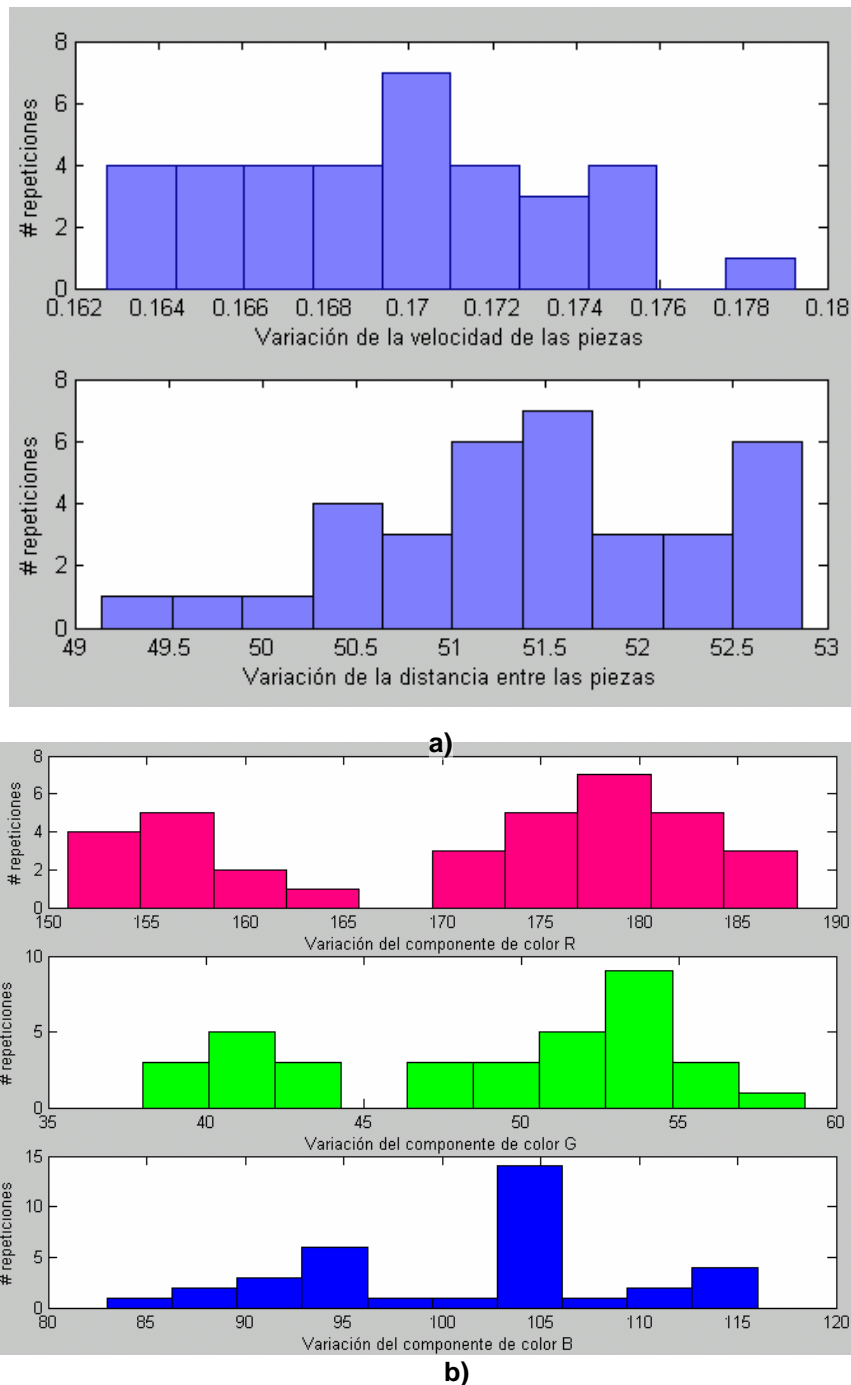
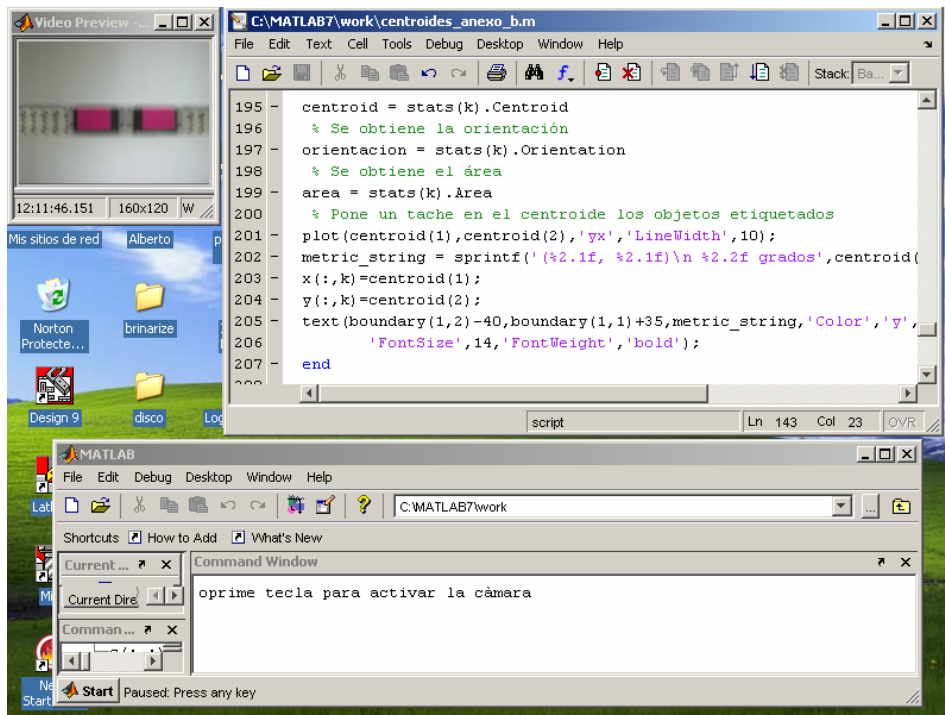
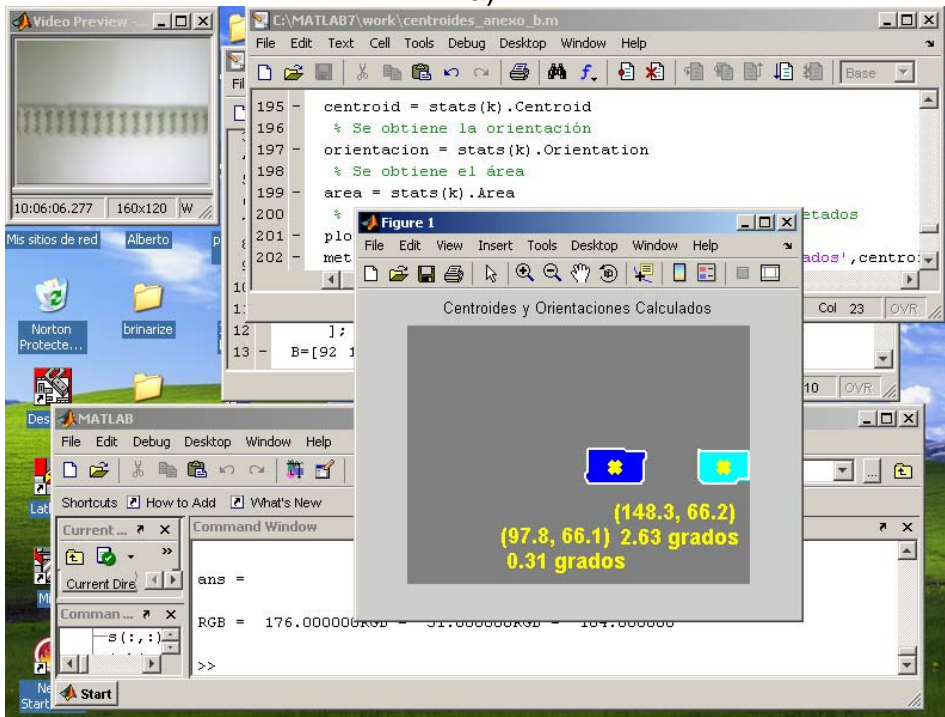


Figura 6.51 a) Histogramas de velocidad, distancia, y b) vector de color RGB.

Es importante, mencionar que el controlador se comunica con la computadora utilizando el protocolo de información RS232 con “HandShaking”, este protocolo permite utilizar más líneas por donde se envían y reciben señales de terminación de envío o recibo de información para agilizar la comunicación.



a)



b)

**Figura 6.52** a) Interfase gráfica, b) corrida de medición de velocidad con el sistema de visión de dos piezas moviéndose sobre la banda

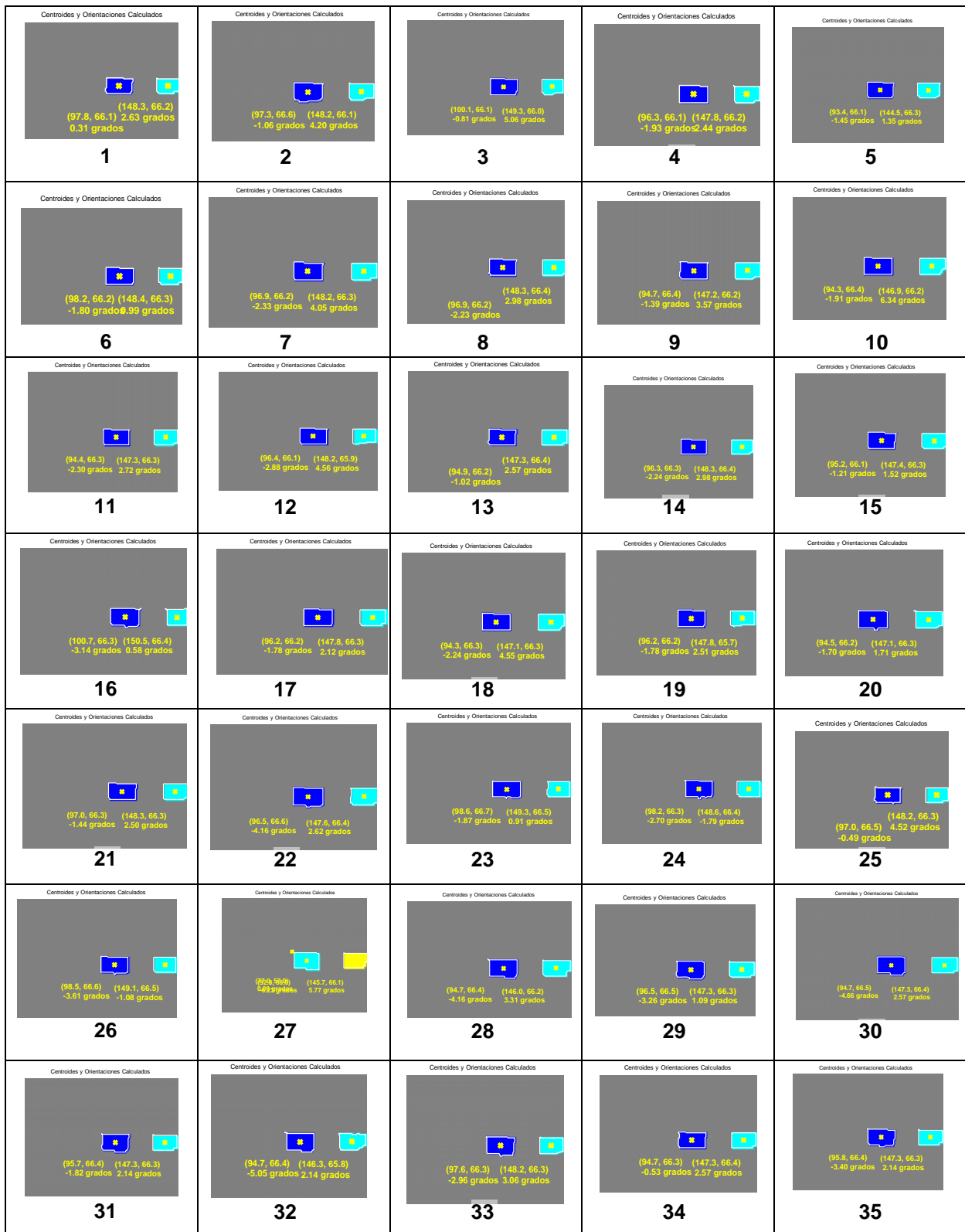
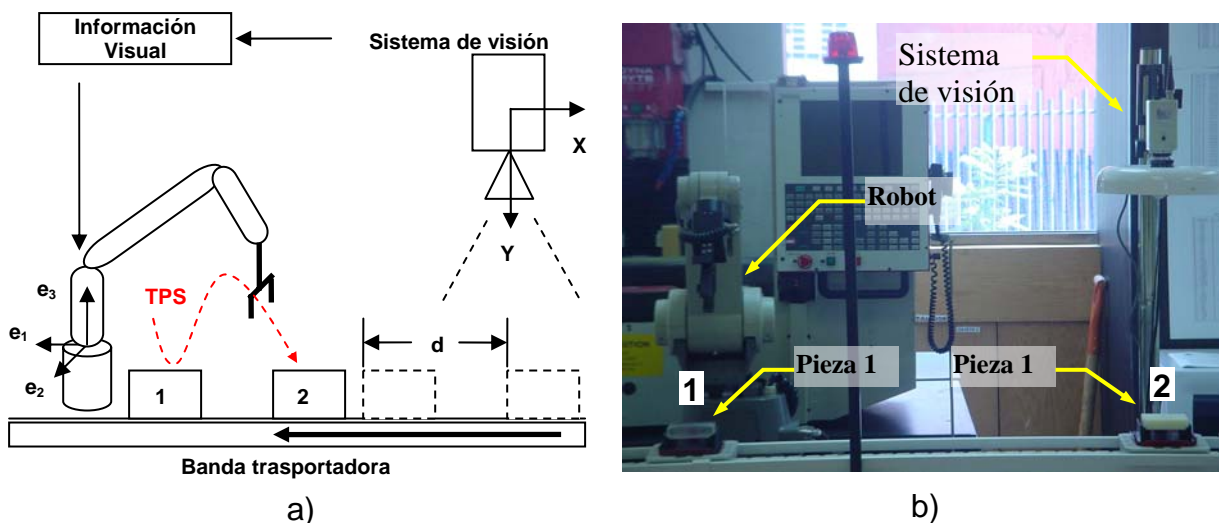


Figura 6.53 Secuencia de eventos realizados para obtención de histograma de velocidad.

## VI.12 PLANTEAMIENTO DEL PROBLEMA PARA LA PARTE EXPERIMENTAL

En los temas anteriores se plantearon las condiciones necesarias para llevar a cabo la parte experimental con el robot Mitsubishi®. El sistema de visión ya se encuentra preparado para medir la distancia entre dos piezas del mismo color y su velocidad. La información anterior, se convierte en datos iniciales para el desarrollo de las funciones seguidoras. En este tema, se plantea el desarrollo de los algoritmos de recuperación de tiempos aplicados a un sistema de manufactura flexible (FMS). En este caso particular, se simula la intercepción de dos piezas que se mueven a lo largo de una banda transportadora utilizando el robot industrial Mitsubishi®, el cual está retroalimentado visualmente con la información de la posición y orientación de las piezas en el tiempo a través de un sistema de visión colocado como se muestra en la figura 6.54.



**Figura 6.54** a) Esquema general del planteamiento del problema, b) montaje.

El montaje experimental se realiza dentro de un Sistema de Manufactura Integrado por Computadora (CIM), se utilizó el robot manipulador Mitsubishi® y el sistema de visión que forman parte del CIM. El módulo es el sistema de manufactura flexible (FMS) y en la parte posterior de la foto se muestra una de las máquinas de CNC que el robot alimenta tomando piezas desde la banda para colocarlas dentro de la máquina. En la misma figura 6.54b, se muestra también la configuración que se planteó para realizar la parte experimental.

### VI.12.1 PLANTEAMIENTO DEL PROBLEMA EXPERIMENTAL:

Sean las piezas 1 y 2 que se mueven a lo largo de una banda transportadora a una distancia  $d = X_1 - X_0$  unidades medida por el sistema de visión. Se pretende generar la trayectoria seguidora TPS con un punto de inicio  $(a_x, a_y)$  que intercepte a las dos piezas dentro del área de trabajo definida por el robot Mitsubishi®, ver figura 6.54. La trayectoria TPS debe igualar la velocidad en los tramos antes estudiados, de tal forma

que no exista impacto al capturar las piezas. Las piezas estarán a una distancia mutua dentro del rango establecido de  $0.45 - 0.15 = 0.30$  m, que es la distancia que el sistema de visión puede detectar.

### VI.12.2 RESTRICCIONES DE TIEMPO PARA EL SISTEMA

a.- En el intervalo de tiempo  $t \leq t_1$  (tiempo de retardo), el sistema de visión adquiere y procesa la información visual para determinar la posición y velocidad de las piezas y la distancia entre ellas. Esta información es utilizada para generar la trayectoria TPS y determinar la cinemática inversa del robot Mitsubishi®.

b.- En el intervalo de tiempo  $t_1 \geq t \leq t_2$  (tiempo de intercepción 1), el robot se mueve para interceptar y capturar la pieza 1.

c.- En el intervalo de tiempo  $t_2 \geq t \leq t_3$  (tiempo de intercepción 2), el robot continua su movimiento hasta interceptar y capturar la pieza 2.

### VI.12.3 RESTRICCIONES FÍSICAS

a.- El efector final debe estar orientado de tal forma que al capturar las piezas no las impacte.

b.- Debe existir un intervalo de tiempo  $\Delta t_1$  para la trayectoria TPL1 y un intervalo de tiempo  $\Delta t_2$  para la trayectoria TPL2, en los cuales la trayectoria TPS iguale las velocidades de las piezas al capturarlas. Es decir, las velocidades relativas entre el efector final de robot al capturar las piezas deben ser cero en los intervalos de tiempo  $\Delta t_1$  y  $\Delta t_2$ .

c.- Las ecuaciones que describen las trayectorias de las piezas están expresadas de la siguiente forma:

$$\text{Pieza 1 TPL1} = \begin{cases} X = x_0 + Vx_0t \\ Y = y_0 + Vy_0t \end{cases} \quad (6.21)$$

$$\text{Pieza 2 TPL2} = \begin{cases} X = x_1 + Vx_1t \\ Y = y_1 + Vy_1t \end{cases} \quad (6.22)$$

Donde  $x_0$  y  $y_0$  son las coordenadas de la posición inicial de la pieza 1 y  $x_1$  y  $y_1$  la posición inicial de la pieza 2, siendo  $V_0$  y  $V_1$  las velocidades de las piezas 1 y 2, respectivamente.

El punto de inicio de la trayectoria seguidora TPS está dada en las coordenadas  $(ax, ay)$ .

### VI.12.4 CONSIDERACIONES CINEMÁTICAS:

a.- En este caso, las velocidades  $V_0$  y  $V_1$  se consideran iguales debido a que las piezas llevan la misma velocidad sobre la banda y se considera también que no existe deslizamiento.

### VI.13 SOLUCIÓN DEL PROBLEMA

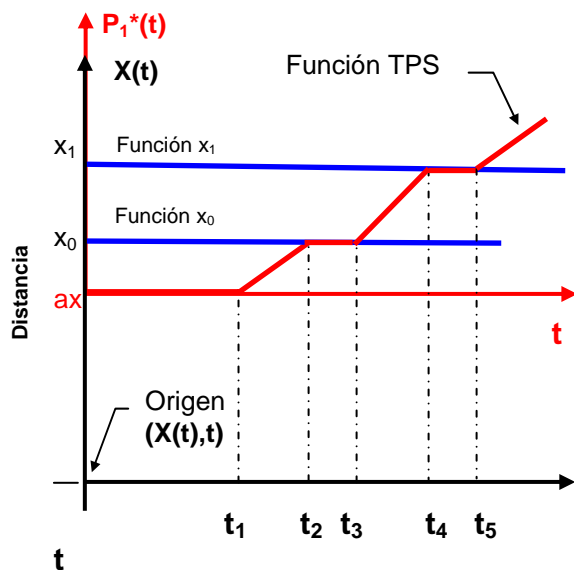
Para solucionar el problema antes planteado, se propone la siguiente metodología:

- Desarrollar el modelo cinemático del robot interceptor.
- Generar la trayectoria TPS.
- Acoplar la función TPS en la cinemática inversa del robot para obtener los ángulos de desplazamiento (en este caso los parámetros de rotación).

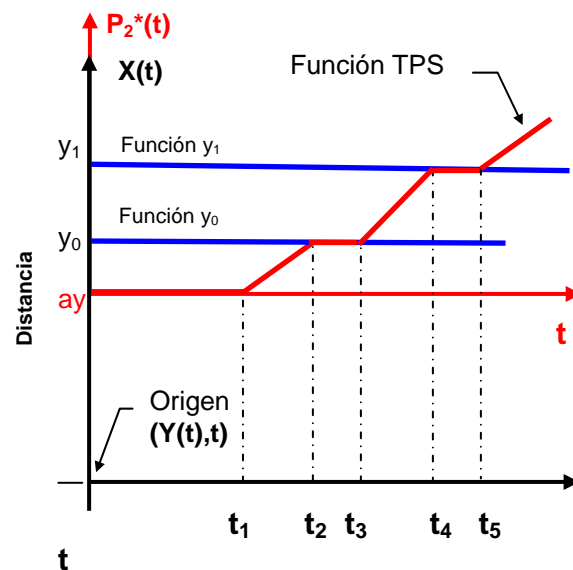
#### VI.13.1 SOLUCIÓN PLANTEADA

a.- El desarrollo del modelo cinemático de un robot industrial Mitsubishi® de 5 gdl, se presentan en las ecuaciones (5.18b, 5.43 y 5.44) del capítulo V. Cabe mencionar, que este modelo se obtuvo utilizando el álgebra de Cuaterniones (capítulo V).

b.- Para generar la trayectoria TPS (algoritmo de recuperación de tiempos), se utilizó la metodología presentada en el tema II.11 del capítulo II, de este trabajo de investigación. Sin embargo, además de obtener la ecuación es necesario igualar las velocidades de las piezas para que el efector final las capture sin impacto. Para ello, se consideran cada uno de los términos de las ecuaciones (6.21 y 6.22) y el punto  $(ax, ay)$  de inicio de la trayectoria seguidora TPS mostradas en las figuras 6.55 a la 6.58 con el fin de plantear los sistemas de ecuaciones:



**Figura 6.55** Funciones  $x_0$  y  $x_1$ , los tiempos en los que la velocidad relativa es cero son  $\Delta t_1 = t_3 - t_2$  y  $\Delta t_2 = t_5 - t_4$ .



**Figura 6.56** Funciones  $y_0$  y  $y_1$ , los tiempos en los que la velocidad relativa es cero son  $\Delta t_1 = t_3 - t_2$  y  $\Delta t_2 = t_5 - t_4$ .

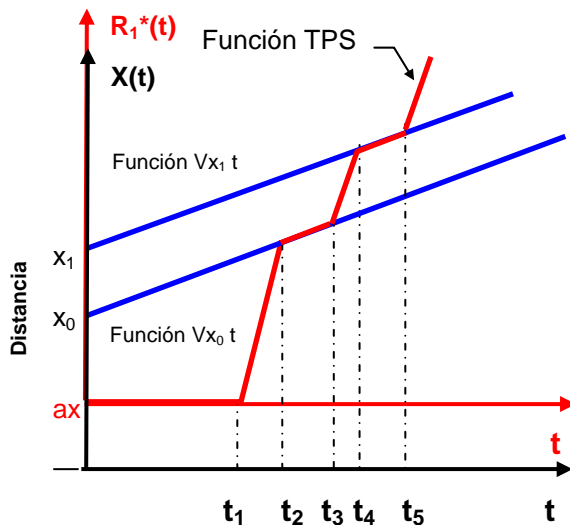


Para calcular  $P_1^*(t)$  se plantea el siguiente sistema de ecuaciones de 5x5.

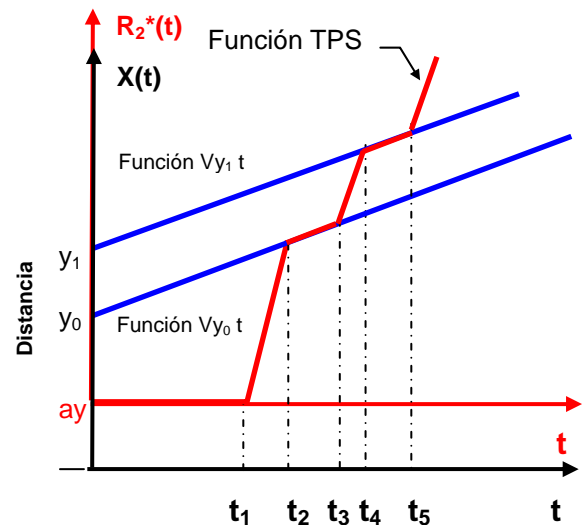
$$\begin{aligned}
 \alpha_0 + \alpha_1 t_1 + \alpha_2 t_1^2 + \alpha_3 t_1^3 + \alpha_4 t_1^4 &= ax & 0 \leq t \leq t_1 \\
 \alpha_0 + \alpha_1 t_2 + \alpha_2 t_2^2 + \alpha_3 t_2^3 + \alpha_4 t_2^4 &= x_0 & t_1 < t \leq t_2 \\
 \alpha_0 + \alpha_1 t_3 + \alpha_2 t_3^2 + \alpha_3 t_3^3 + \alpha_4 t_3^4 &= x_0 & t_2 < t \leq t_3 \\
 \alpha_0 + \alpha_1 t_4 + \alpha_2 t_4^2 + \alpha_3 t_4^3 + \alpha_4 t_4^4 &= x_1 & t_3 < t \leq t_4 \\
 \alpha_0 + \alpha_1 t_5 + \alpha_2 t_5^2 + \alpha_3 t_5^3 + \alpha_4 t_5^4 &= x_1 & t_4 < t \leq t_5
 \end{aligned} \tag{6.23}$$

Para calcular  $P_2^*(t)$  se plantea el siguiente sistema de ecuaciones de 5x5.

$$\begin{aligned}
 \beta_0 + \beta_1 t_1 + \beta_2 t_1^2 + \beta_3 t_1^3 + \beta_4 t_1^4 &= ay \\
 \beta_0 + \beta_1 t_2 + \beta_2 t_2^2 + \beta_3 t_2^3 + \beta_4 t_2^4 &= y_0 \\
 \beta_0 + \beta_1 t_3 + \beta_2 t_3^2 + \beta_3 t_3^3 + \beta_4 t_3^4 &= y_0 \\
 \beta_0 + \beta_1 t_4 + \beta_2 t_4^2 + \beta_3 t_4^3 + \beta_4 t_4^4 &= y_1 \\
 \beta_0 + \beta_1 t_5 + \beta_2 t_5^2 + \beta_3 t_5^3 + \beta_4 t_5^4 &= y_1
 \end{aligned} \tag{6.24}$$



**Figura 6.57** Funciones  $V_{x_0}t$  y  $V_{x_1}t$ , los tiempos en los que la velocidad relativa es cero son  $\Delta t_1=t_3-t_2$  y  $\Delta t_2=t_5-t_4$ .



**Figura 6.58** Funciones  $V_{y_0}t$  y  $V_{y_1}t$ , los tiempos en los que la velocidad relativa es cero son  $\Delta t_1=t_3-t_2$  y  $\Delta t_2=t_5-t_4$ .

Para calcular  $R_1^*(t)$  se plantea el siguiente sistema de ecuaciones de 5x5.

$$\begin{aligned}
 \chi_0 + \chi_1 t_1 + \chi_2 t_1^2 + \chi_3 t_1^3 + \chi_4 t_1^4 &= 0 \\
 \chi_0 + \chi_1 t_2 + \chi_2 t_2^2 + \chi_3 t_2^3 + \chi_4 t_2^4 &= V_{x_0} t_2 \\
 \chi_0 + \chi_1 t_3 + \chi_2 t_3^2 + \chi_3 t_3^3 + \chi_4 t_3^4 &= V_{x_0} t_3 \\
 \chi_0 + \chi_1 t_4 + \chi_2 t_4^2 + \chi_3 t_4^3 + \chi_4 t_4^4 &= V_{x_1} t_4 \\
 \chi_0 + \chi_1 t_5 + \chi_2 t_5^2 + \chi_3 t_5^3 + \chi_4 t_5^4 &= V_{x_1} t_5
 \end{aligned} \tag{6.25}$$

Para calcular  $R_2^*(t)$  se plantea el siguiente sistema de ecuaciones de 5x5.

$$\begin{aligned}
 \delta_0 + \delta_1 t_1 + \delta_2 t_1^2 + \delta_3 t_1^3 + \delta_4 t_1^4 &= 0 \\
 \delta_0 + \delta_1 t_2 + \delta_2 t_2^2 + \delta_3 t_2^3 + \delta_4 t_2^4 &= Vy_0 t_2 \\
 \delta_0 + \delta_1 t_3 + \delta_2 t_3^2 + \delta_3 t_3^3 + \delta_4 t_3^4 &= Vy_0 t_3 \\
 \delta_0 + \delta_1 t_4 + \delta_2 t_4^2 + \delta_3 t_4^3 + \delta_4 t_4^4 &= Vy_1 t_4 \\
 \delta_0 + \delta_1 t_5 + \delta_2 t_5^2 + \delta_3 t_5^3 + \delta_4 t_5^4 &= Vy_1 t_5
 \end{aligned} \tag{6.26}$$

Finalmente se construyen las funciones de la siguiente forma:

$$P_1^*(t) = \alpha_0 + \alpha_1 t + \alpha_2 t^2 + \alpha_3 t^3 + \alpha_4 t^4 \tag{6.27}$$

$$P_2^*(t) = \beta_0 + \beta_1 t + \beta_2 t^2 + \beta_3 t^3 + \beta_4 t^4 \tag{6.28}$$

$$R_1^*(t) = \chi_0 + \chi_1 t + \chi_2 t^2 + \chi_3 t^3 + \chi_4 t^4 \tag{6.29}$$

$$R_2^*(t) = \delta_0 + \delta_1 t + \delta_2 t^2 + \delta_3 t^3 + \delta_4 t^4 \tag{6.30}$$

$$TPS = TPS_x + TPS_y \tag{6.31}$$

donde,

$$TPS_x = P_1^*(t) + R_1^*(t)$$

$$TPS_y = P_2^*(t) + R_2^*(t)$$

Cabe aclarar, que en este caso particular el componente  $R_2^*(t)=0$ , debido a que las piezas se mueven paralelamente al eje X. Es decir, sobre la banda transportadora.

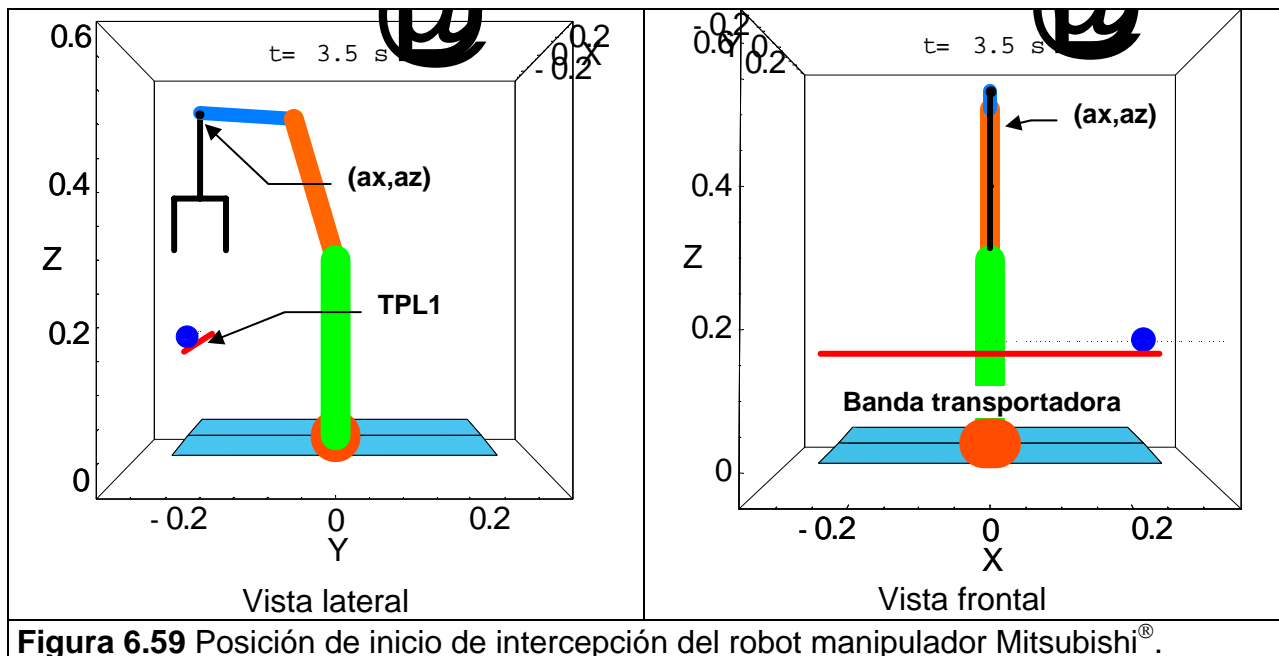
## VI.14 ANÁLISIS DE LA TRAYECTORIA TPS

En este tema se analiza la función TPS generada a partir de la información del sistema de visión, que en este caso fue: la velocidad de las piezas sobre la banda es de 0.17 m/s aprox. y la distancia  $d = 0.25$  m entre ellas.

La tabla 6.19 muestra los datos utilizados para realizar este evento experimental. Cabe aclarar que en este caso, para coincidir con el sistema de coordenadas del robot Mitsubishi®,  $az = ay = 0.55$  m, la posición  $(ax, az)$  corresponde al extremo final del eslabón 3, que es el inicio de movimiento de la trayectoria seguidora TPS, ver figura 6.59. La distancia entre las piezas es:  $d = 0.25$  m.

**Tabla 6.19**  
 Datos iniciales y definición de tiempos de intercepción

Pieza #1 TPL1			Pieza #2 TPL2		
Posición inicial:	$X_0 =$	0.75 m	Posición inicial:	$X_1 =$	1.00 m
	$Y_0 =$	0.43 m		$Y_1 =$	0.43 m
Velocidad:	$V_0 =$	-0.17 m/s	Velocidad:	$V_1 =$	-0.17 m/s
Tiempo de retardo:	$t_1 =$	3.0 s	Tiempo de retardo:	$t_1 =$	3.0 s
Tiempo de intercepción 1:	$t_2 =$	4.0 s	Tiempo de intercepción 2:	$t_4 =$	8.0 s
Tiempo $\Delta t$	$\Delta t =$	0.5 s	Tiempo $\Delta t$	$\Delta t =$	0.5 s
Tiempo para atrapar	$t_3 =$	4.5 s	Tiempo para atrapar	$t_5 =$	8.5 s
Punto de inicio de TPSx	$a_x =$	-0.23 m	Punto de inicio de TPSx	$a_x =$	-0.23 m
Punto de inicio de TPSy	$a_z =$	0.55 m	Punto de inicio de TPSy	$a_z =$	0.55 m



**Figura 6.59** Posición de inicio de intercepción del robot manipulador Mitsubishi®.

Con los valores iniciales de la tabla 6.19 se obtiene para la trayectoria TPS:

**Posición:**

$$\begin{aligned}
 TPS_x &= -9.24857 + 7.21012 t - 1.97004 t^2 + 0.22619 t^3 - 0.00936508 t^4 \\
 TPS_y &= 3.99073 - 2.54545 t + 0.658182 t^2 - 0.0727273 t^3 + 0.00290909 t^4 \quad (6.32)
 \end{aligned}$$

**Velocidad:**

$$\begin{aligned}
 VTPS_x &= 7.21012 - 3.94008 t + 0.678571 t^2 - 0.0374603 t^3 \\
 VTPS_y &= -2.54545 + 1.31636 t - 0.218182 t^2 + 0.0116364 t^3 \quad (6.33)
 \end{aligned}$$

**Aceleración:**

$$\begin{aligned}
 ATPS_x &= -3.94008 + 1.35714 t - 0.112381 t^2 \\
 ATPS_y &= 1.31636 - 0.436364 t + 0.0349091 t^2 \quad (6.34)
 \end{aligned}$$

En la figura 6.60 a) y b), se puede observar como la función TPS iguala la posición y velocidad de las funciones TPL1 y TPL2 en sus intervalos definidos por los tiempos de intercepción  $t_2 = 4.5$  s y  $t_4 = 8.0$  s, respectivamente. También se presentan con flechas los rangos de tiempo  $\Delta t$ . En el rango de tiempo  $\Delta t_1$  la función TPSx toca la curva TPL1x y casi iguala su posición dentro del rango establecido de 0.5 s para este caso particular, después TPSx toca la función TPL2x en el rango de tiempo  $\Delta t_2$  y es evidente que el error que en este caso es la diferencia entre la distancia de las funciones mencionadas es un poco mayor. Sin embargo, cabe señalar que como se trata de funciones no se debe olvidar que a un valor del dominio sólo y únicamente le corresponde uno del contradominio respetándose el teorema fundamental de las funciones.

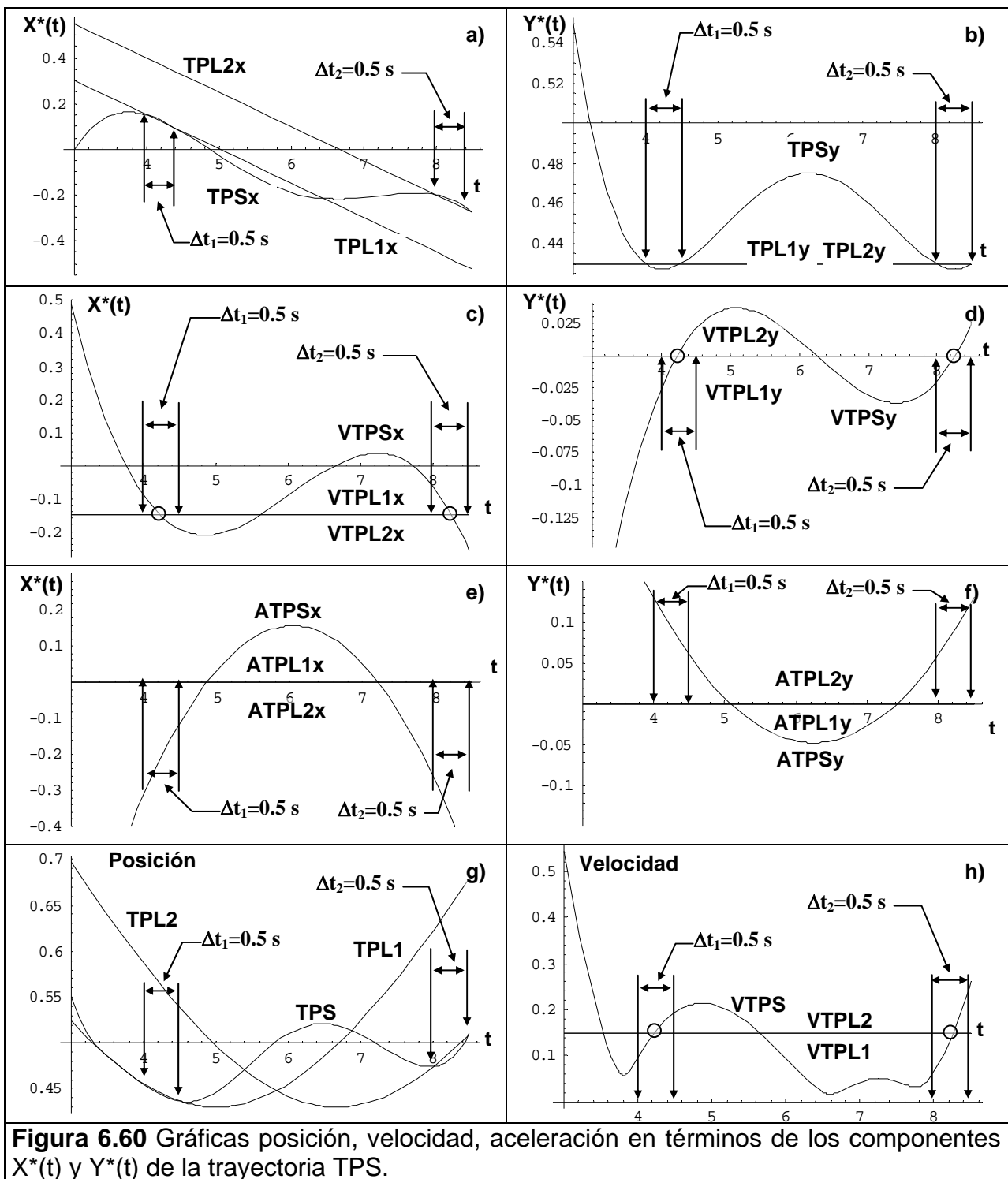
No obstante, la aproximación a la curva TPL1x y TPL2x por la función TPSx es aceptable para la aplicación de nuestro caso experimental de estudio. En la figura 6.6b ocurre el mismo fenómeno solo que en este caso se estudia el eje  $Y^*(y)$ , aquí la función TPSy toca dos puntos de cada una de las funciones TPL1y y TPL2y en sus componente  $Y^*(t)$ , esto también sucede en el eje  $X^*(t)$  pero no se alcanza a observar en la gráfica.

Es importante mencionar que la posición de las piezas no varía a lo largo del eje  $y(t)$  debido a que las piezas solo se mueven a lo largo del eje  $X(t)$ , es decir, sobre la banda.

En la figura 6.60c se analiza la velocidad de las funciones VTPL1x, VTPL2x y VTPSx en su componente  $X(t)$ , respectivamente. Se puede observar que la velocidad relativa entre las trayectorias mencionadas en los rangos de tiempo  $\Delta t_1$  y  $\Delta t_2$  no es cero, esto significa que solo en el punto de intercepción marcado con un círculo la velocidad de las funciones se iguala, pero usando el mismo criterio anterior, el error oscila entre  $-0.1$  y  $-0.2$  aproximadamente, y en la aplicación del evento experimental es permisible. Lo mismo sucede para el análisis en el eje  $Y^*(t)$ , solo que aquí la velocidad de las piezas es cero, ver figura 6.60d.

Con respecto a la aceleración, es natural que sea mayor la aceleración de la partícula seguidora TPS porque necesita alcanzar las partículas libres TPL1 y TPL2 en el mismo intervalo de tiempo, aunque la aceleración de las partículas libres sea casi cero en la realidad.

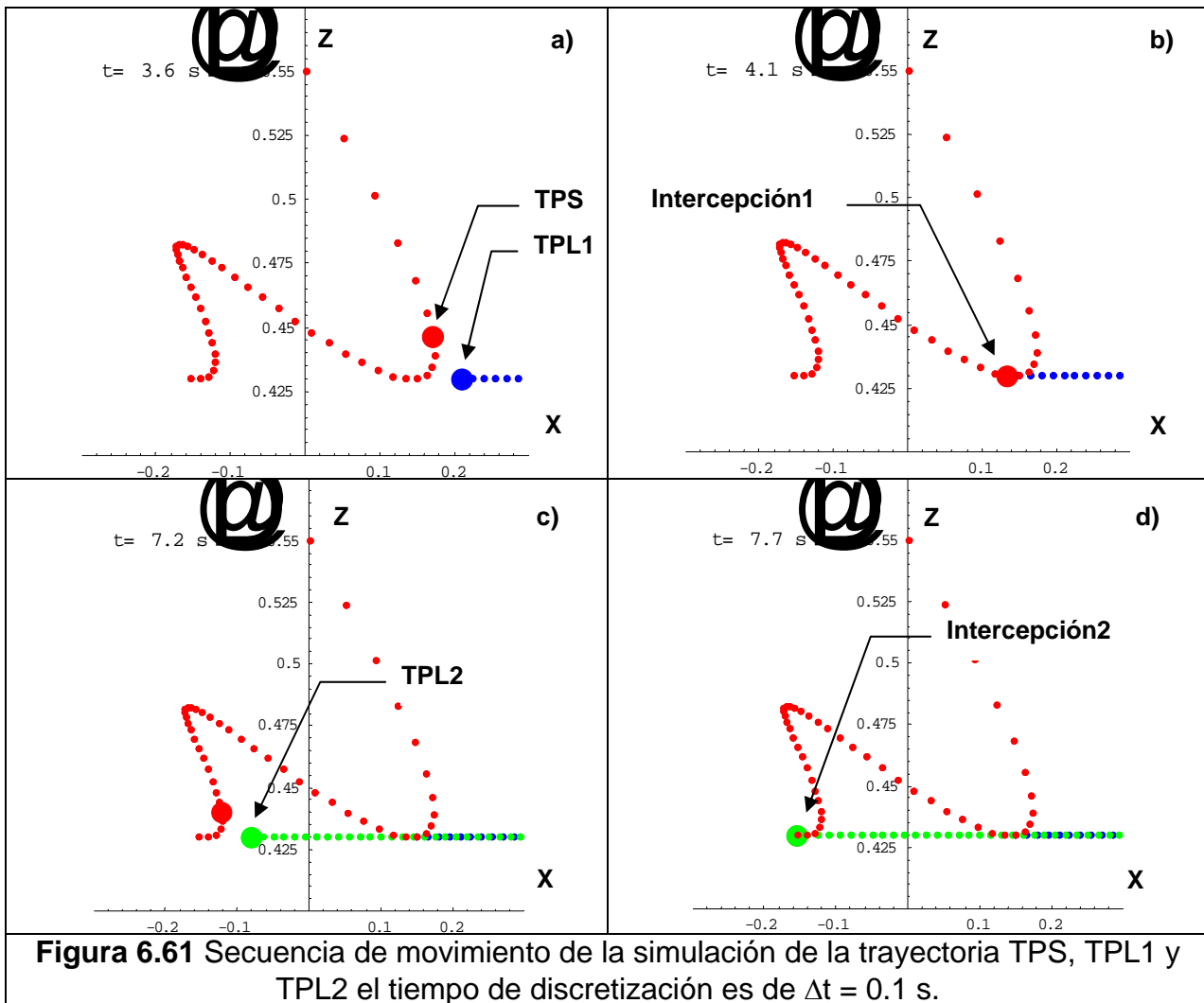
Finalmente, se presenta en la figura 6.60g y h las magnitudes de las funciones para corroborar lo anteriormente descrito, se vuelve a presentar la misma diferencia en la distancia de posición de las trayectorias y la velocidad de la misma magnitud para las tres funciones solo en el punto medio.



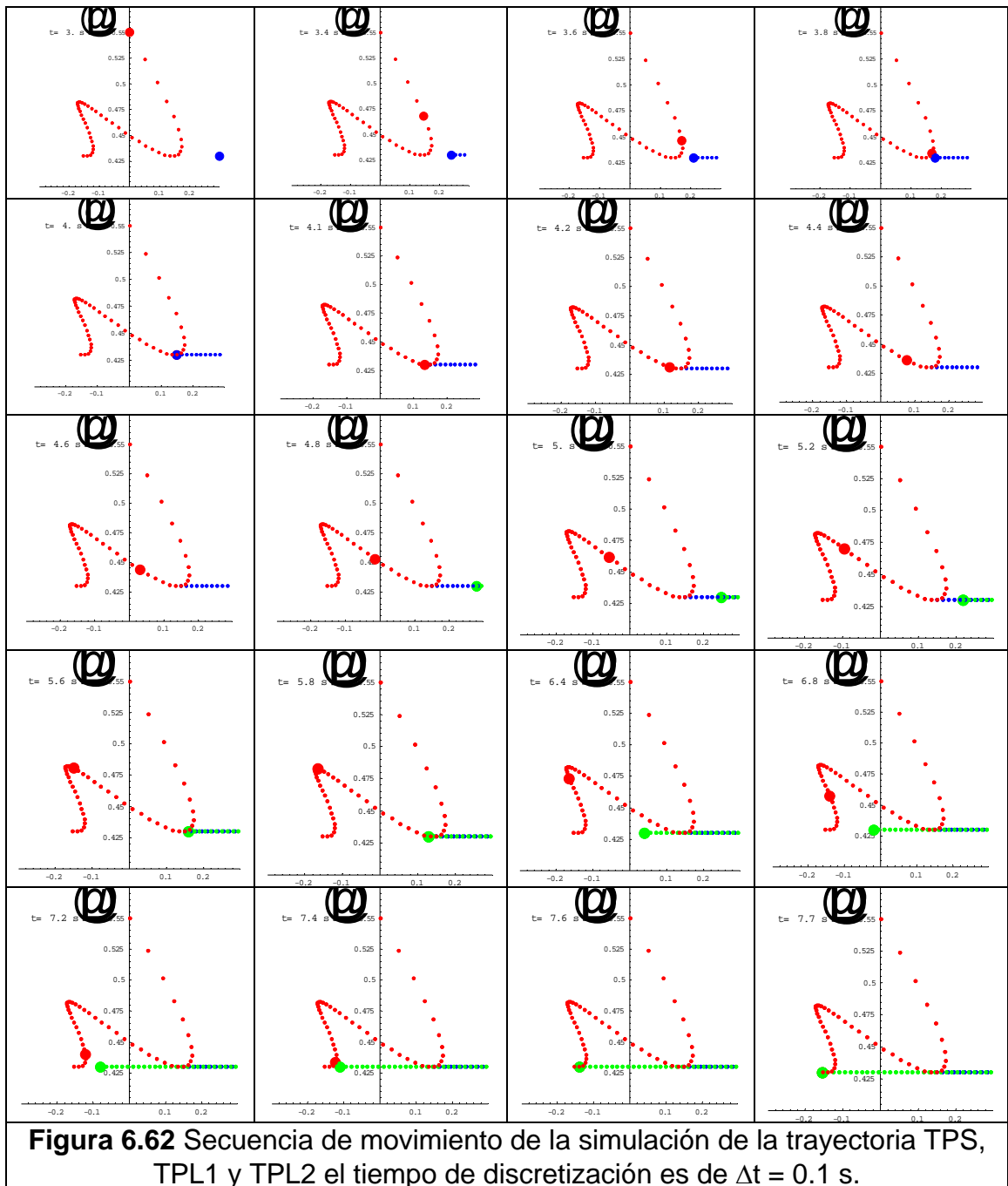
**Figura 6.60** Gráficas posición, velocidad, aceleración en términos de los componentes  $X^*(t)$  y  $Y^*(t)$  de la trayectoria TPS.

La gráfica de la figura 6.60g se obtuvo utilizando la ecuación (6.35) en el intervalo de tiempo  $t_1$  a  $t_5$  s, la figura 6.60h y g muestran la magnitud de la posición y velocidad respectivamente de las funciones TPL1, TPL2 y TPS. De lo anterior, se puede concluir que para reducir el error en la velocidad de seguimiento de la trayectoria TPS en los instantes de tiempo  $\Delta t_1$  y  $\Delta t_2$  se debe disminuir el tiempo de  $\Delta t$ , dependiendo de las necesidades de operación y de las restricciones de tiempo del sistema en sí.

$$\sqrt{TPL1x^2 + TPL1y^2}, \sqrt{TPL2x^2 + TPL2y^2}, \sqrt{TPSx^2 + TPSy^2} \quad (6.35)$$



En la figura 6.62 se muestra la secuencia completa de imágenes desde el tiempo de retardo  $t_1$  hasta el tiempo de intercepción 2 (tiempo  $t_4$ ). Lo anterior permitirá programar la secuencia de movimiento de los eslabones del robot a través del modelo cinemático inverso.



Así mismo en la figura 6.62 se presenta la secuencia de intercepción múltiple con algunas gráficas, el intervalo de tiempo fue de  $\Delta t = 0.1$  s. Sin embargo, este intervalo puede definirse más grande o pequeño arbitrariamente. A continuación se presenta la simulación del fenómeno físico utilizando un  $\Delta t = 0.5$  s. En la figura 6.63 se presentan algunas imágenes de la secuencia de la trayectorias TPS, TPL1 y TPL2 donde se observa que TPS intercepta a TPL1 en el tiempo  $t = 4.0$  s e iguala y mantiene la velocidad de la partícula durante  $\Delta t_1 = 0.5$  s. Finalmente, con la misma trayectoria TPS se encuentra a TPL2 y la intercepta en el tiempo  $t = 8.0$  e iguala y mantiene la velocidad

de la partícula durante  $\Delta t_2=0.5$  s. Esto debido al planteamiento de las ecuaciones (6.32). Es importante mencionar, que el número de los tiempos  $\Delta t$  en los cuales se iguala la velocidad de las partículas puede ser mayor, esto es;  $n\Delta t$ , donde  $n$  es el número de incrementos de tiempo  $\Delta t$ . este tiempo  $\Delta t$  está en función del tiempo que necesita el gripper para cerrarse y atrapar la pieza. Debe notarse que el tiempo total de intercepción  $t_5$  creció de 7.7 s a 8.5 s, y el incremento de la distancia, ver figuras 6.61 y 6.63g a la h, con un  $\Delta t = 0.5$ s.

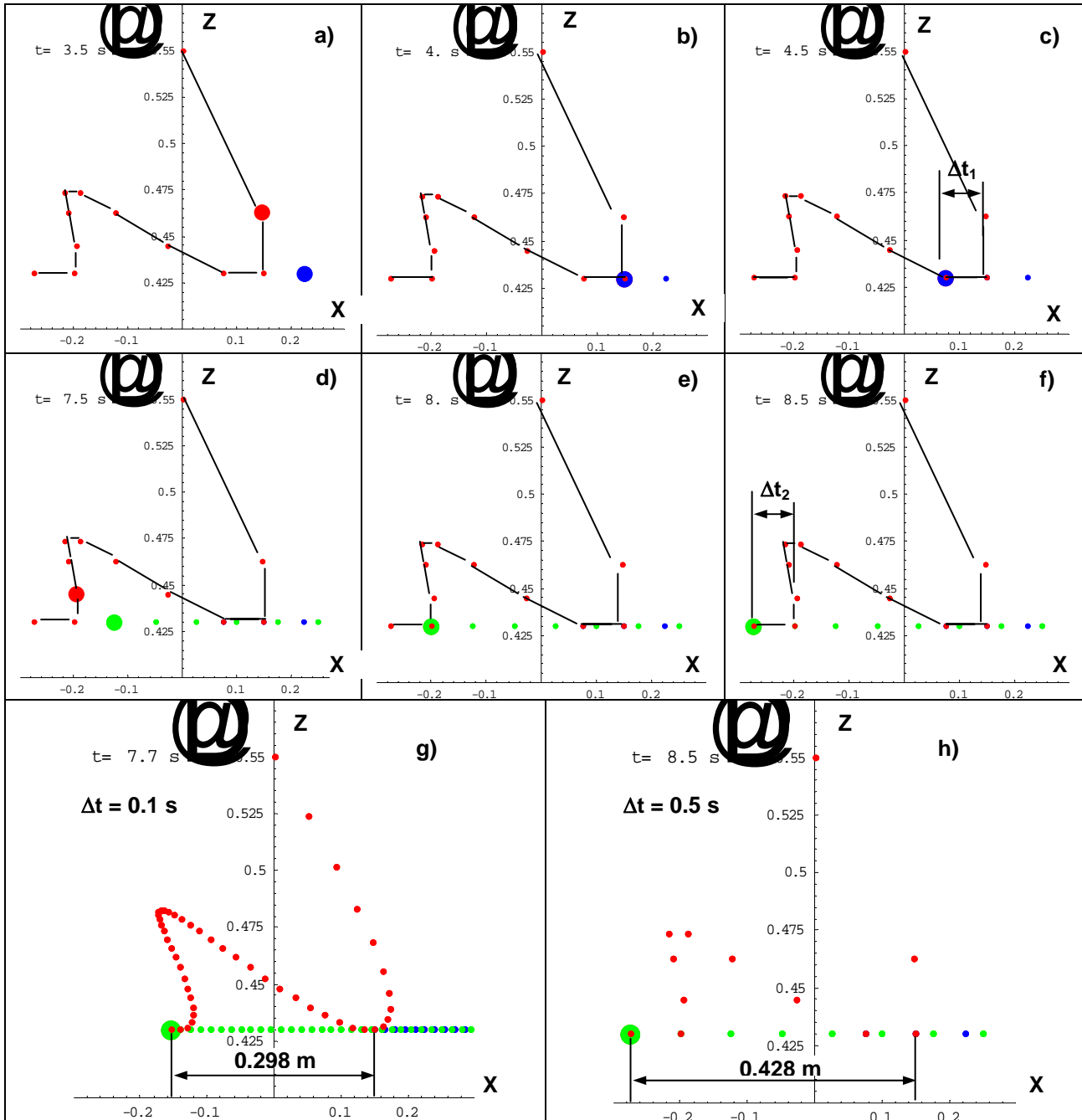
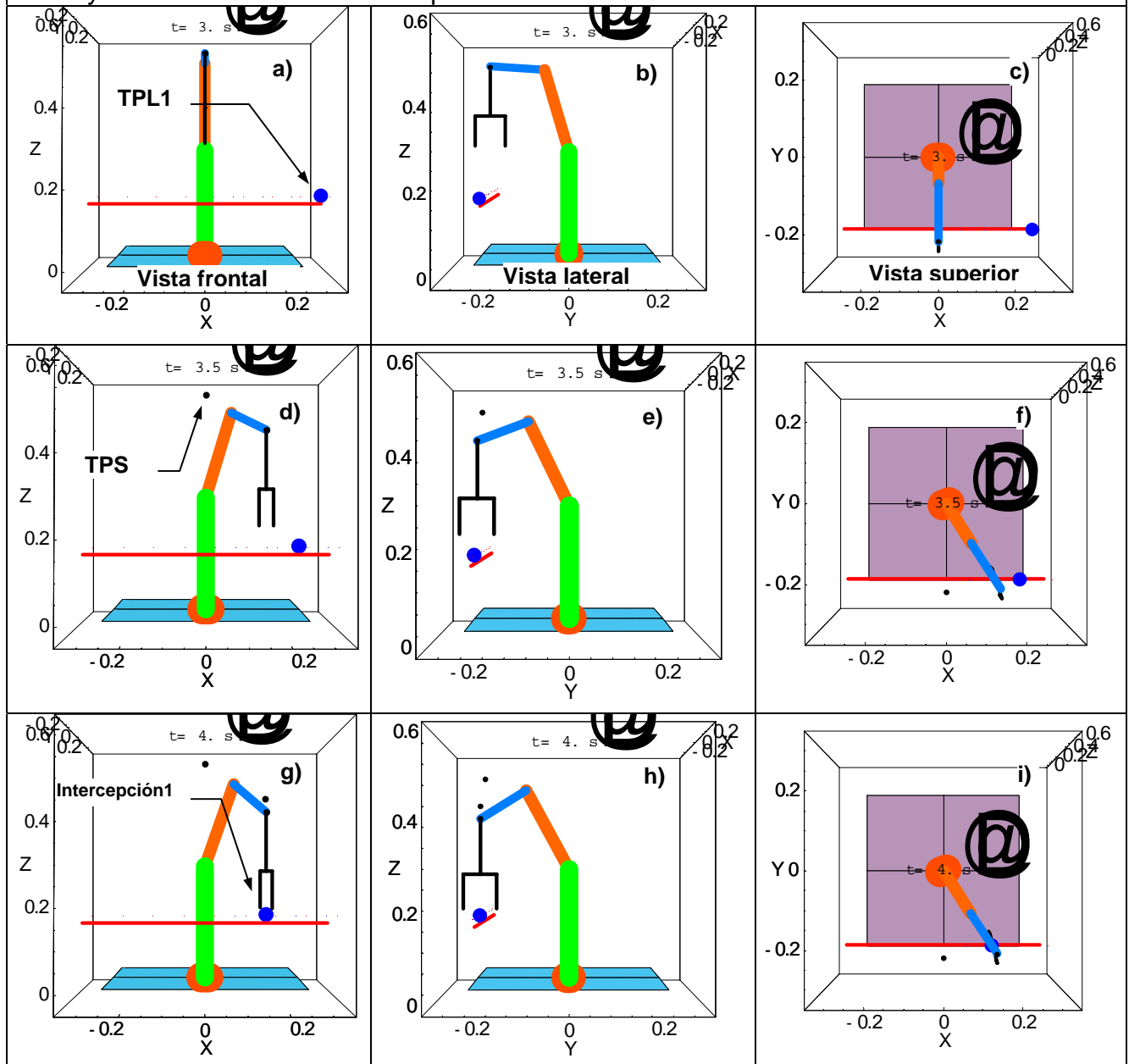


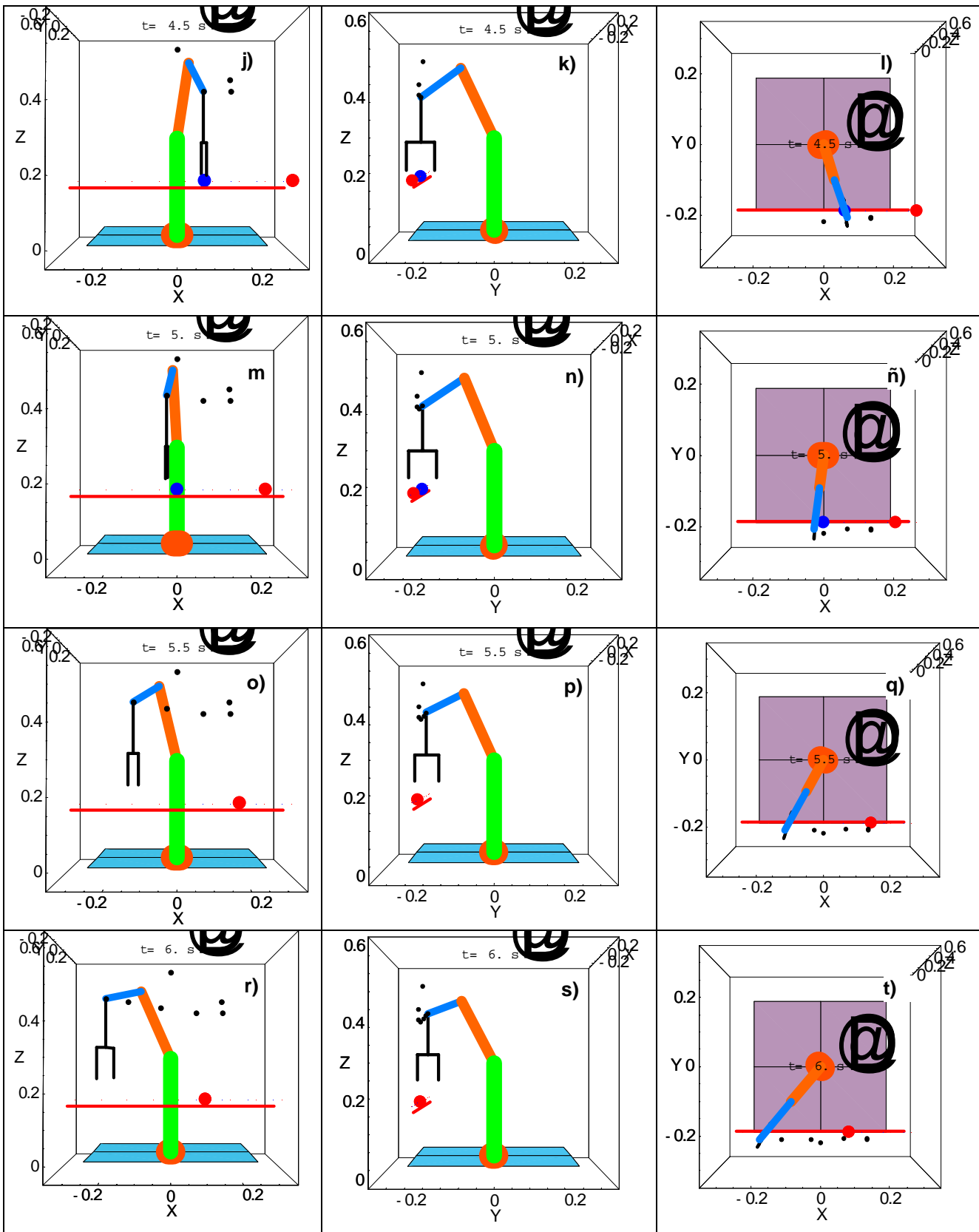
Figura 6.63 Secuencia de la trayectoria TPS en el rango  $t_1=3.0$  a  $t_5=8.5$  s con  $\Delta t=0.5$  s

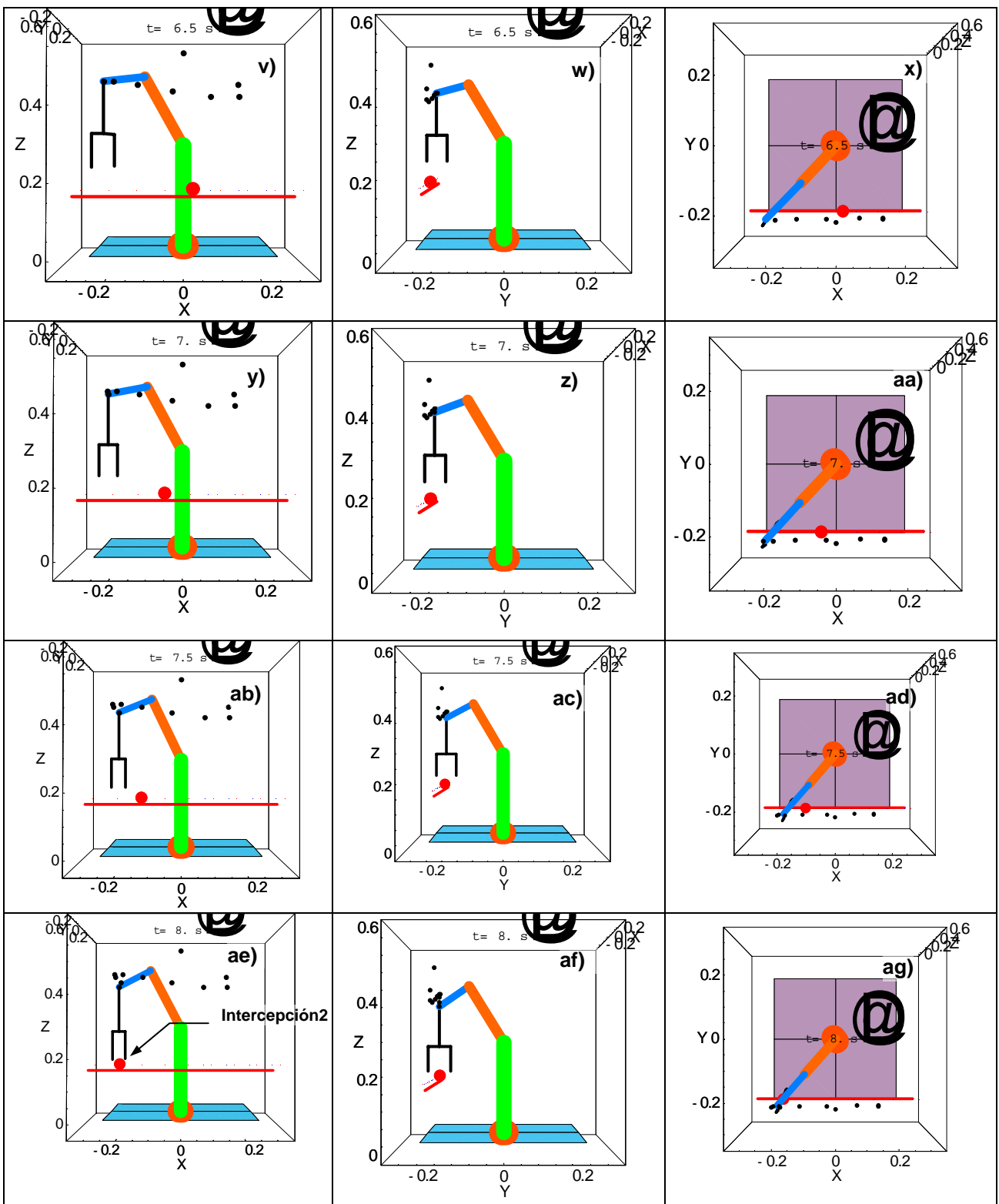


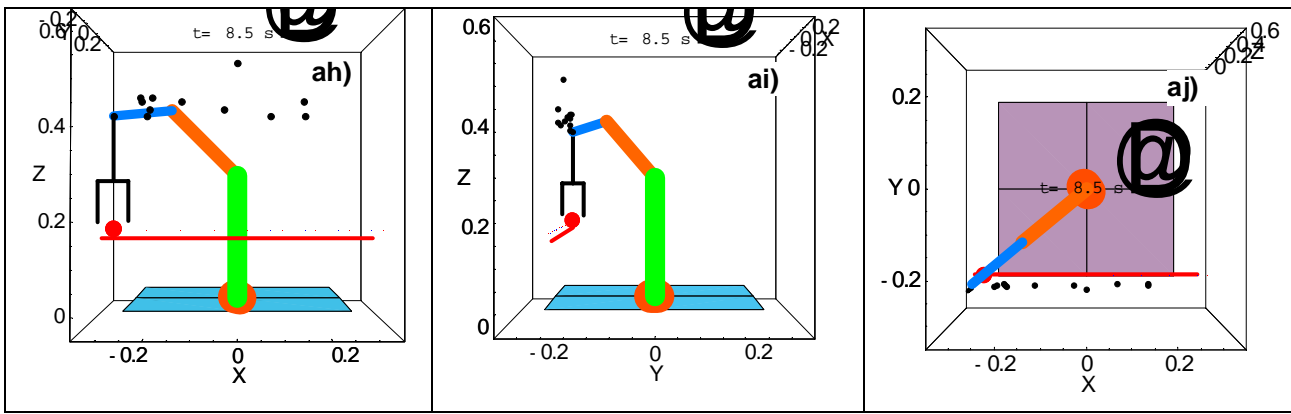
c.- El acoplamiento de la trayectoria TPS a la ecuación (5.43), generó las siguientes gráficas, ver tabla 6.20. La figura 6.20b es simplemente la vista lateral donde se observa que el efector final se orienta de tal forma que no impacta la partícula. Cabe mencionar, que la partícula representa el centro de gravedad de algún objeto, que para este caso particular de aplicación es un cubo con dimensiones 8 x 5 x 2 cm.

**Tabla 6.20** Secuencia de movimiento del robot Mitsubishi® 5 gdl con la trayectoria TPS, TPL1 y TPL2 con intervalo de tiempo  $\Delta t = 0.5$  s.









De lo anterior se decidió que el evento de intercepción se hará con una trayectoria TPS obtenida utilizando un intervalo de tiempo  $\Delta t = 0.5$  s, debido a que es el tiempo necesario para que la pinza del robot industrial cierre y atrape a la pieza.

### VI.15 COORDENADAS ARTICULARES DEL ROBOT MITSUBISHI®

En este tema, se obtienen las funciones que representan la evolución en el tiempo de las coordenadas del robot  $\theta_1(t)$ ,  $\theta_2(t)$ ,  $\theta_3(t)$ ,  $\theta_4(t)$ ,  $\theta_5(t)$  y sus velocidades angulares respectivamente.

En la tabla 6.21 se muestran las posiciones del efector final del robot dentro del área de trabajo y los valores de las variables articulares de posición.

Tabla 6.21 Valores numéricos de las coordenadas articulares absolutas									
n	$\Delta t$	teta1	teta2	teta3	teta4	teta5	X	Y	Z
1	3	0	0	0	93.6168	90	0	-0.23	0.55
2	3.5	32.4827	13.0001	10.8031	69.8135	102.391	0.146429	-0.23	0.462727
3	4	33.1113	16.5991	16.9436	60.074	108.023	0.15	-0.23	0.43
4	4.5	18.0605	10.2957	29.0693	54.2518	100.786	0.075	-0.23	0.43
5	5	-7.02279	6.35097	29.5179	57.7479	86.2389	-0.0283333	-0.23	0.445273
6	5.5	-28.2937	10.5347	15.2143	67.8677	78.5357	-0.12381	-0.23	0.462727
7	6	-39.4537	18.0631	-2.67414	78.2278	80.4686	-0.189286	-0.23	0.473636
8	6.5	-43.2902	22.661	-11.6261	82.5818	83.0654	-0.216667	-0.23	0.473636
9	7	-42.6551	22.2972	-6.93813	78.2577	79.3804	-0.211905	-0.23	0.462727
10	7.5	-40.2921	20.8016	2.54862	70.2665	74.0254	-0.195	-0.23	0.445273
11	8	-41.0091	22.8703	4.21439	66.532	70.8994	-0.2	-0.23	0.43
12	8.5	-50.0921	36.715	-25.8183	82.7201	81.3846	-0.275	-0.23	0.43

En la figura 6.64 se muestran las posiciones que el efector final debe seguir según la tabla 6.21.

En las figuras 6.65 a la 6.66 se presentan las posiciones con respecto al tiempo que debe ir alcanzando el efector final del robot para seguir la trayectoria de las piezas sobre la banda e interceptarlas.

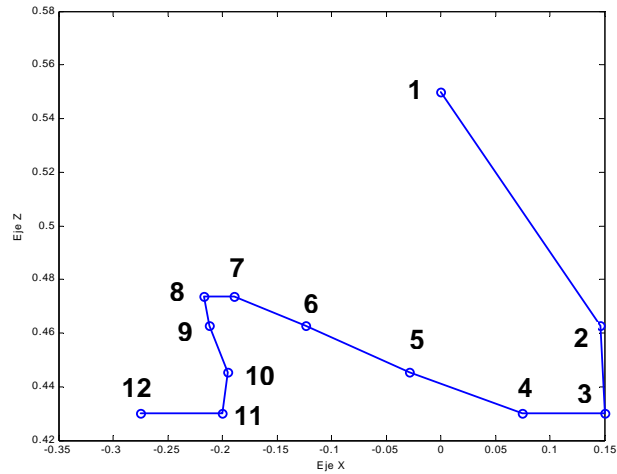


Figura 6.64 Posiciones de la trayectoria TPS

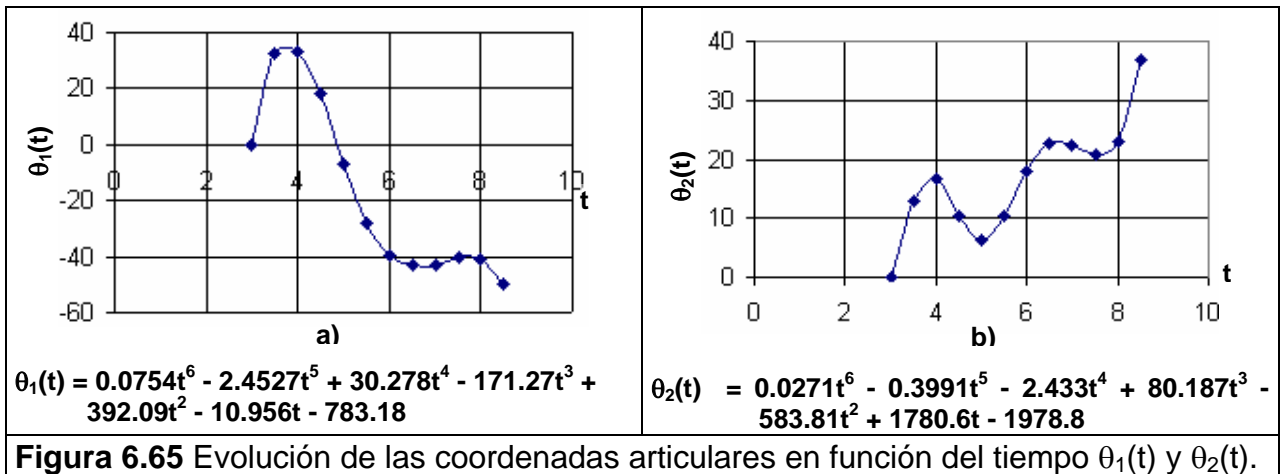


Figura 6.65 Evolución de las coordenadas articulares en función del tiempo  $\theta_1(t)$  y  $\theta_2(t)$ .

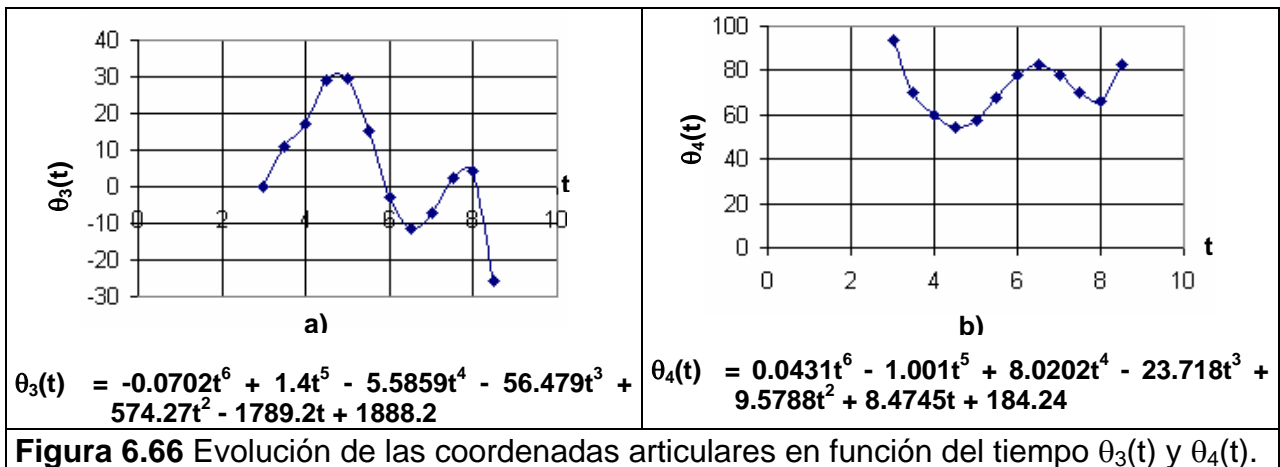
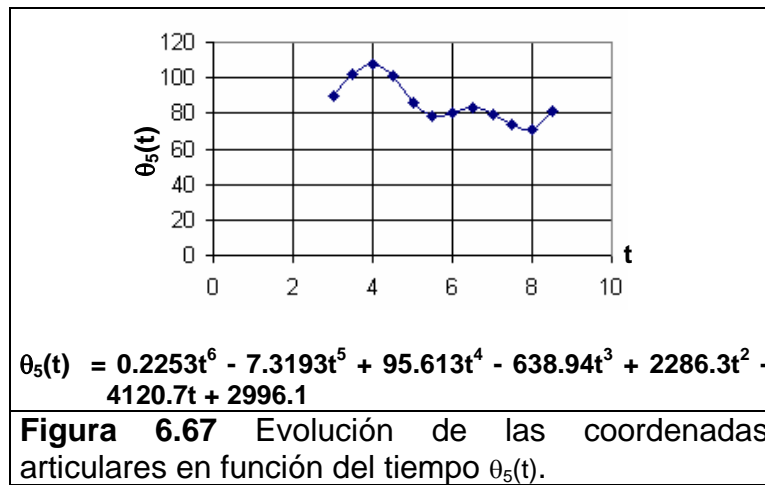


Figura 6.66 Evolución de las coordenadas articulares en función del tiempo  $\theta_3(t)$  y  $\theta_4(t)$ .



**Tabla 6.22** Valores numéricos de las coordenadas articulares relativas

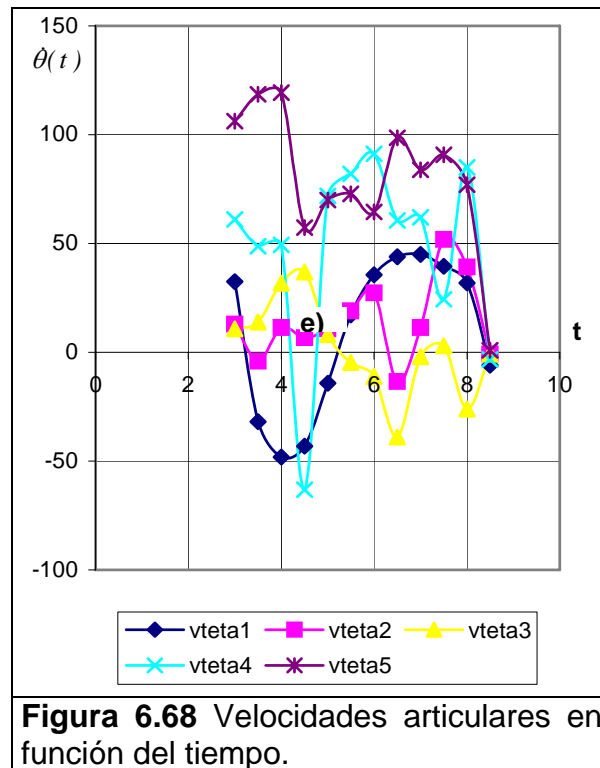
n	$\Delta t$	rteta1	rteta2	rteta3	rteta4	rteta5
1	3	0	0	0	0	0
2	3.5	32.4827	13.0001	10.8031	-23.8033	12.391
3	4	0.6286	3.599	6.1405	-9.7395	5.632
4	4.5	-15.0508	-6.3034	12.1257	-5.8222	-7.237
5	5	-25.08329	-3.94473	0.4486	3.4961	-14.5471
6	5.5	-21.27091	4.18373	-14.3036	10.1198	-7.7032
7	6	-11.16	7.5284	-17.88844	10.3601	1.9329
8	6.5	-3.8365	4.5979	-8.95196	4.354	2.5968
9	7	0.6351	-0.3638	4.68797	-4.3241	-3.685
10	7.5	2.363	-1.4956	9.48675	-7.9912	-5.355
11	8	-0.717	2.0687	1.66577	-3.7345	-3.126
12	8.5	-9.083	13.8447	-30.03269	16.1881	10.4852

En la tabla 6.22, se presentan los valores de las coordenadas articulares que se programaran para generar la secuencia de movimiento del robot Mitsubishi®, cabe aclarar que los movimientos que ejecuta el robot industrial deben estar dados en coordenadas relativas.

**Tabla 6.23** Valores numéricos de las velocidades articulares

n	t	vteta1	vteta2	vteta3	vteta4	vteta5
1	3	32.4827	13.0001	10.8031	61.1477652	106.171988
2	3.5	-31.8541	-4.08193723	13.9419051	48.704649	118.535963
3	4	-48.1621	11.3985716	31.7573096	49.2975293	119.33964
4	4.5	-43.14379	6.76143051	36.8870482	-63.1879003	57.4107772
5	5	-14.24812	10.8332754	8.15889729	71.9049983	70.0139012
6	5.5	17.1337	19.0070695	-4.69506078	82.0217418	72.8880071
7	6	35.6172	27.3692236	-11.0444997	91.3204226	64.5838658
8	6.5	43.9253	-13.3837951	-38.8955242	60.6420156	98.5902703
9	7	45.0181	11.3656288	-2.09040781	62.017342	83.9588768
10	7.5	39.5751	51.8822944	2.98239889	24.3494058	90.7214378
11	8	31.9261	39.2329236	-26.1227046	84.9354194	77.0048766
12	8.5	-5.89318824	-0.73294991	-0.7032085	-3.20393287	0.98385519
promedio		<b>32.4149582</b>	<b>17.4207666</b>	<b>15.6735054</b>	<b>58.5610935</b>	<b>80.0169549</b>

Se puede observar que el máximo desplazamiento angular lo tiene el eslabón 1, que es la base del robot, y el mínimo el eslabón 5, que es la pinza, con esa trayectoria se orienta para atrapar la piezas sin impacto.

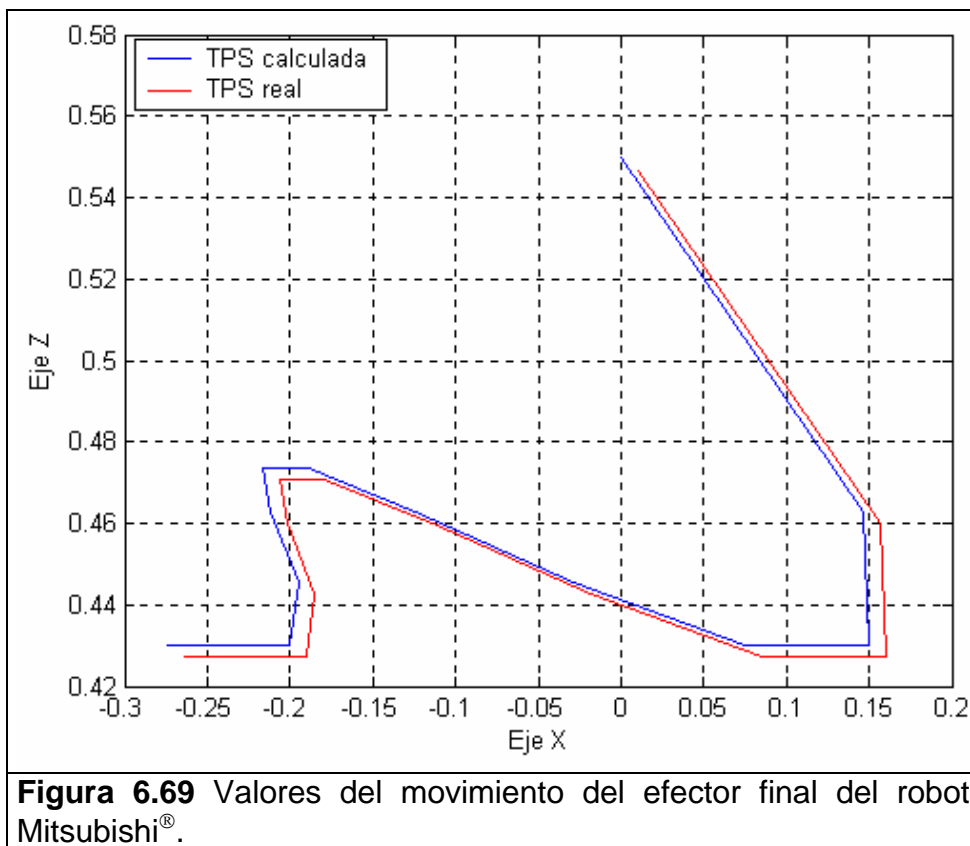


$$\begin{aligned}
 \theta_1(t) &= 0.4524t^5 - 12.2635t^4 + 121.112t^3 - 513.81t^2 + 784.18t - 10.956 \\
 \theta_2(t) &= 0.1626t^5 - 1.9955t^4 - 9.732t^3 + 240.561t^2 - 1167.62t + 1780.6 \\
 \theta_3(t) &= -0.4212t^5 + 7.0t^4 - 22.3436t^3 - 169.437t^2 + 1148.54t - 1789.2 \\
 \theta_4(t) &= 0.2586t^5 - 5.005t^4 + 32.0808t^3 - 71.154t^2 + 19.1576t + 8.4745 \\
 \theta_5(t) &= 1.3518t^5 - 36.5965t^4 + 382.452t^3 - 1916.82t^2 + 4572.6t - 4120.7
 \end{aligned}
 \tag{6.36}$$

En la figura 6.68 se presentan las cinco funciones de posición y tiempo de los eslabones del robot, y en la ecuación (6.36) se presentan las funciones de cada eslabón, éstas serán programadas en programa de control como un elemento más para llevar a cabo la parte experimental.

## VI.16 RESULTADOS EXPERIMENTALES DEL MOVIMIENTO DEL EFECTOR FINAL DEL ROBOT MITSUBISHI®.

En la figura 6.69 se presenta las posiciones leídas desde el buffer del controlador del robot Industrial y se comparan con las posiciones calculadas. El error máximo es 0.0034 m y el mínimo de 0.0026 m en el eje Z, y para el eje X el mínimo fue de 0.0100 m y el máximo de 0.013 m. Los errores anteriores son debido a la exactitud y precisión del robot que es de  $\pm 2$  grados (0.3 mm en coordenadas cartesianas). Sin embargo, no afectó en la trayectoria del robot para capturar las piezas.



**Figura 6.69** Valores del movimiento del efector final del robot Mitsubishi®.

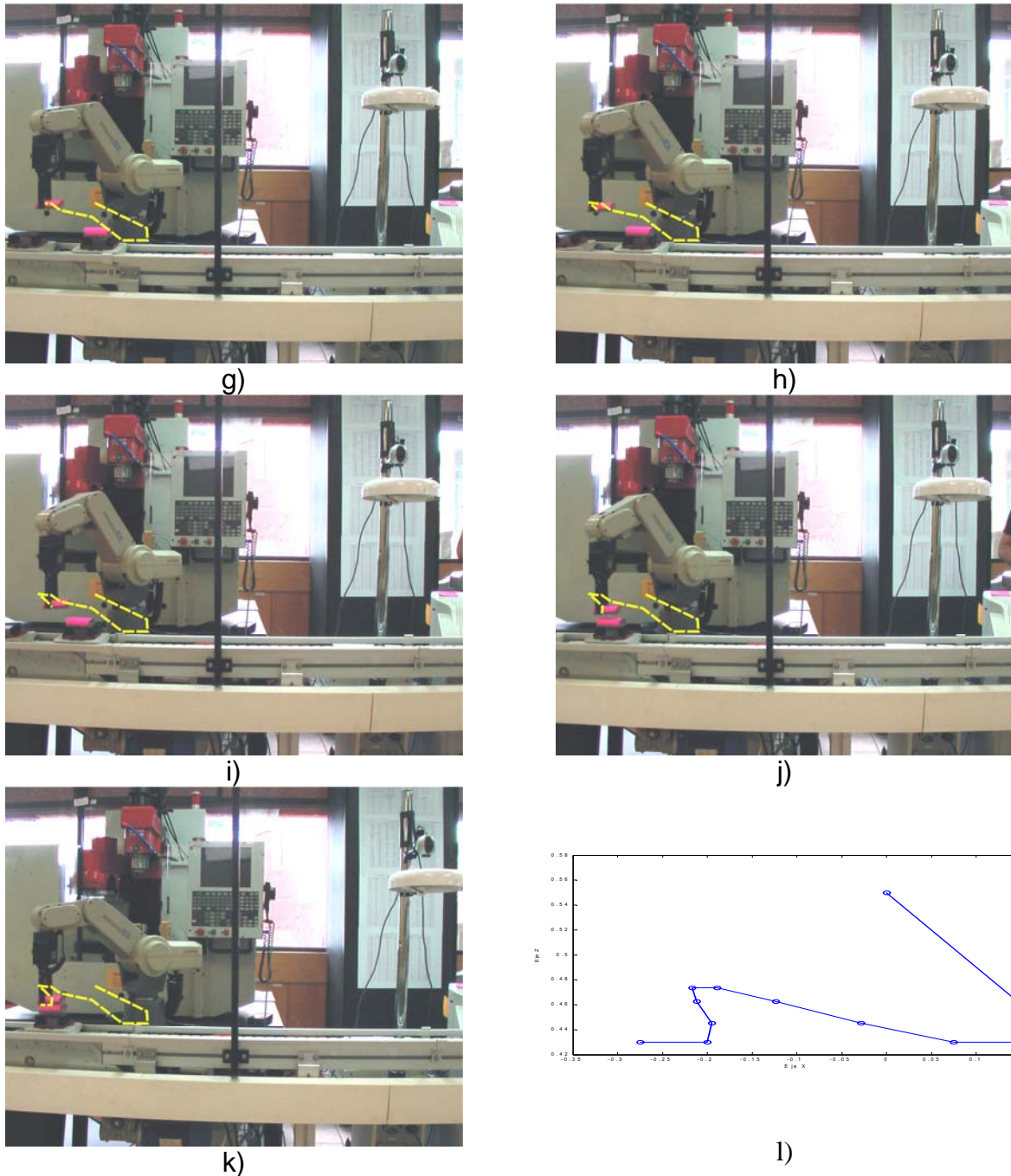
A continuación se muestran algunas fotografías del evento experimental que se realizó, ver figura 6.70. En la secuencia presentada se dibujó la trayectoria que el robot iba ejecutando. La trayectoria TPS (real) tiene una diferencia notoria con respecto a la calculada y esto es debido a que el robot industrial tiene una precisión de posición de 0.3 mm lo que significa que siempre existirá un error, el cual evidentemente es de índole no corregible, por tal motivo, la trayectoria es diferente a la calculada. Sin embargo, lo anterior permitió de todas formas capturar las piezas con cierto margen de error. Es importante mencionar que de cada 10 eventos realizados 7 fueron exitosos, los 3 restantes tuvieron fallos en el tiempo de generación de resultados por la computadora, es decir, el cálculo de la velocidad y distancia entre las piezas, ya que las restricciones de tiempo que se plantearon no fueron totalmente cubiertas aún utilizando



el programa con la opción prioridad de tarea en “tiempo real”, sin embargo, es una excelente aproximación para emplear un sistema operativo que trabaje en tiempo real.



**Figura 6.70** Sequencia de imágenes del evento experimental en el sistema (FMS).

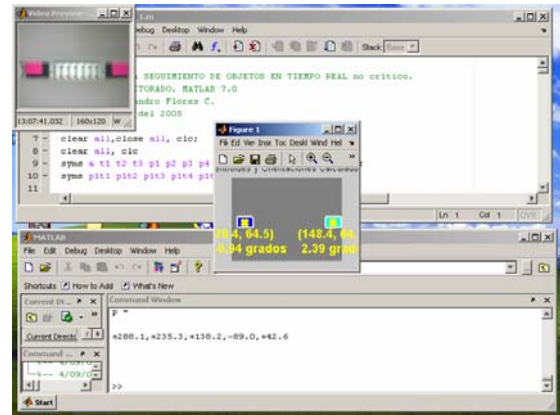


**Figura 6.70** Secuencia de imágenes del evento experimental en el sistema (FMS).

En la tabla 6.24 se presentan los tiempos que requirió el sistema de visión y el robot Mitsubishi® para realizar sus respectivas tareas. El área del sistema de visión es de 0.45 m a lo largo de la banda, esa distancia la recorrieron las dos piezas que viajan sobre la banda separadas una distancia de 0.365 m (36.5 cm). El tiempo de cada proceso es el que utilizó el sistema de visión para capturar la imagen, y realizar el procesamiento necesario para obtener el centroide y la orientación de cada una de las piezas con el fin de calcular la velocidad y la distancia de separación entre ellas. Las posiciones de la 1 a la 11 que siguió el robot fueron alcanzadas en intervalos de tiempo de  $\Delta t = 0.5$  seg, ver la misma tabla.



a)



b)



c)



d)

**Figura 6.71** El sistema de visión en operación.

En la figura 6.71 se presenta el sistema de visión en operación, este montaje consta de una base y una lámpara para iluminar la escena de operación por donde pasan las piezas sobre la banda transportadora, cuenta con un dispositivo mecánico que permite el deslizamiento de la cámara para acercarse o alejarse de la banda según sea el caso.

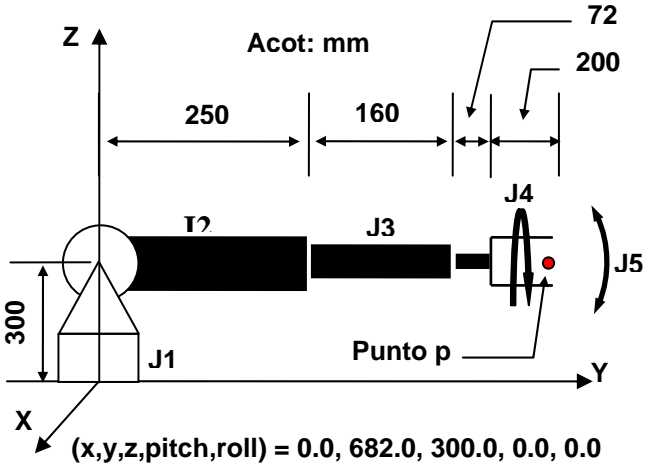
El sistema de visión funcionó correctamente, cuando las piezas entran dentro del área de visión se dispara automáticamente el proceso para capturar las imágenes y realizar el procesamiento de la información visual y obtener la velocidad de las piezas.

En la tabla 6.24 se muestran los resultados y los tiempos que el sistema de visión necesitó para realizar la adquisición de la imagen y procesarla. En el evento experimental el sistema de visión logró capturar 25 tomas durante un tiempo de 2.533 seg, y estas tomas fueron suficientes para calcular la velocidad de las piezas y su posición relativa entre ellas. Además, en la misma tabla 6.24 se presentan las coordenadas leídas desde el buffer del controlador del robot las cuales tienen un pequeño margen de error mostrado en la figura 6.69. la posición de "Home" es la posición desde la cual el robot mide todos los desplazamientos angulares de sus eslabones, la posición "preparatoria" es una posición que define el usuario dependiendo de las condiciones y restricciones del entorno del sistema. En la figura 6.70a se muestra la posición de inicio de intercepción que es la posición desde la cual se miden los

desplazamientos angulares presentados en la secuencia de imágenes ver figura 6.70a...k.

Tabla 6.24 Tiempos requeridos en el evento experimental			
Sistema de visión		Robot Mitsubishi®	
# cuadro	Tiempo requerido por cuadro acumulado en seg.	# posición y movimiento del robot	Posición real del robot en coordenadas: (x, y, z, wrist pich, wrist roll) grados (°)
1	0.100000	Home	(+96.6,-167.4,+754.7,+10.9,+179.9)
2	0.190000	preparatoria	(0.0,+193.3,+754.7,+10.9,-8.0)
3	0.290000	Inicio de intercepción	(0.0,+325.0,+222.1,-89.0,-8.0)
4	0.380000	1	(-161.0,+252.8,+117.9,-89.0,-40.4)
5	0.481000	2	(-161.0,+252.8,+117.9,-89.0,-40.4)
6	0.571000	3	(-80.1,+255.7,+122.7,-89.0,-25.4)
7	0.691000	4	(+32.7,+242.6,+169.8,-89.0,-0.3)
8	0.781000	5	(+132.7,+239.7,+186.6,-89.0,+20.9)
9	0.881000	6	(+199.9,+236.8,+196.8,-89.0,+32.1)
10	0.992000	7	(+227.4,+235.7,+196.3,-89.0,+35.8)
11	1.092000	8	(+222.7,+235.7,+185.6,-89.0,+35.3)
12	1.192000	9	(+205.5,+236.5,+168.2,-89.0,+32.9)
13	1.292000	10	(+212.7,+239.0,+141.3,-89.0,+33.6)
14	1.382000	11	(+288.1,+235.3,+138.2,-89.0,+42.6)
15	1.512000		
16	1.612000		
17	1.702000		
18	1.802000		
19	1.922000		
20	2.032000		
21	2.133000		
22	2.233000		
23	2.323000		
24	2.433000		
25	2.533000		



**Figura 6.72** Origen mecánico del robot  
 posición del punto p (x,y,z,pitch,roll) = ( 0.0, 682.0, 300.0, 0.0, 0.0 ).

Los resultados que arrojó el sistema de visión fueron los siguientes: El vector de color promedio fue: RGB (208, 43, 129) para ambas piezas color rosa, para la pieza 1 el centroide se localizó en: x=27.0920 y=64.9632 píxeles y la orientación fue: -3.8679° y su área: 326 píxeles<sup>2</sup>. Para la pieza 2, el centroide se localizó en: x=149.7083 y=65.6354 píxeles, su orientación = 1.9093° y su área = 384 píxeles<sup>2</sup>. La velocidad de ambas piezas fue de 0.175681 m/s, y la distancia entre ellas d=122.616309 píxeles. El tiempo que requirió el robot para capturar las piezas fue de 8.53 seg. El tiempo total del

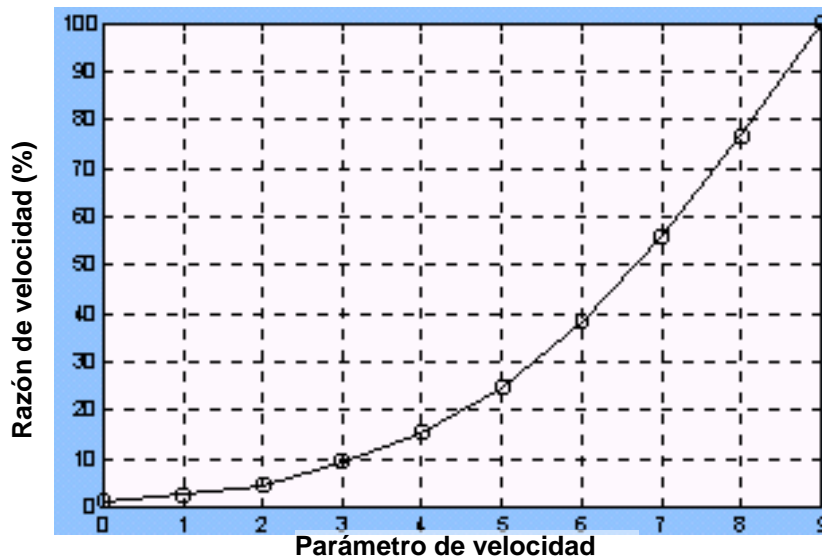
evento fue de:  $2.5333 + 8.53 + 3.2 = 14.2633$  seg. Cabe aclarar que las piezas llegaron hasta el área de trabajo del robot Mitsubishi® en 3.2 s.

Tabla 6.25 Especificaciones del robot Mitsubishi®.			
dato	Especificaciones	Comentarios	
<b>Estructura mecánica</b>			
<b>5 grados de libertad. Robot articulado vertical</b>			
Rango de operación	Rotación base	300° (máx. 120°/seg)	Eje J1
	Rotación hombro	130° (máx. 72°/seg)	Eje J2
	Rotación del codo	110° (máx. 109°/seg)	Eje J3
	Inclinación de muñeca	±90° (máx. 100°/seg)	Eje J4
	Rotación de muñeca	±180° (máx. 163°/seg)	Eje J5
Capacidad de carga	1.2 kg (incluyendo el peso del efector final)		
Velocidad máxima del efector final	1000 mm/seg (superficie de herramienta de muñeca)		
Precisión en posición	0.3 mm		
Sistema de movimiento	Eléctrico: servomotores DC		
Capacidad de los servomotores	J1 a J3: 30W; J4, J5: 11W.		
Peso del robot	19 kg aproximadamente.		

En la tabla 6.25 se muestran las especificaciones del fabricante ofrecidas en el robot industrial, en donde se aprecian los límites de velocidad de cada uno de los eslabones. Cabe mencionar, que la velocidad de cada eslabón por lo regular se establece dentro de un rango de porcentaje de la velocidad del eslabón que tiene más pulsos en un movimiento del robot determinado. Por lo que es necesario trabajar con porcentajes de velocidad y no con cantidades numéricas fijas como 30°/seg, 67.3°/seg, etc [6.3].

Lo anterior dificulta introducir los valores obtenidos de velocidad en el programa de control en forma directa, por lo que se acarreo un error inherente en la generación de la trayectoria. Otra limitante que se observó, fue que las velocidades solo se pueden introducir como velocidades desde el número "0" hasta el número "9" en términos de porcentaje de velocidad, ver figura 6.73, y esta restricción obligó a realizar aproximaciones a uno de los valores de 0 a 9 y no a las velocidades reales. Lo anterior, se reflejo en un incremento en el error de la trayectoria calculada. Es importante mencionar que este tipo de problema al programar velocidades en robot industriales limita su campo de aplicación.

Cabe mencionar que muchos robots industriales tienen la misma forma de considerar las velocidades angulares, por ejemplo, el robot industrial CRS® A-465 de la empresa CRS Robotics Corporation [6.4], también trabaja con porcentajes de velocidad pero permite realizar incrementos de 1 en 1 grados, por lo que la resolución es mayor comparada con la del robot Mitsubishi®. Una posible alternativa para un trabajo futuro sería utilizar robot CRS® para implementar los algoritmos de recuperación de tiempos.



SP	0	1	2	3	4	5	6	7	8	9
%	1.2	2.4	4.7	9.4	15.3	24.7	38.3	56.0	76.5	100

**Figura 6.73** Relación de parámetros de rapidez y velocidad del robot industrial.

En la figura 6.73 se muestra la relación de velocidades fijas con las que puede operar el robot Mistubishi® en porcentaje con respecto al eslabón que tiene la máxima velocidad angular.

Cuando dos o más ejes están en operación el parámetro de velocidad válido que se toma en consideración para realizar los cálculos es el del eslabón que tiene el máximo número de pulsos. Es importante mencionar que la velocidad especificada no se podrá alcanzar si la distancia de movimiento es más pequeña que la distancia de aceleración/desaceleración dada.

Lo anterior, restringió la velocidad a la cual debe moverse el robot para que intercepte las dos piezas debido a que para resolver este problema se aproximaron las velocidades calculadas a las establecidas por el fabricante.

## VI.17 SUMARIO

Se llevaron a cabo los dos experimentos propuestos para comprobar y validar experimentalmente los algoritmos de recuperación de tiempos. Se construyó un montaje y un prototipo de robot de dos grados de libertad con un gripper en forma de canasta para capturar dos esferas en caída libre, y se diseñó y construyó toda la electrónica y las conexiones necesarias para medir la posición y la velocidad de los eslabones del robot interceptor.

En el caso del robot de 2 gdl, se estudió un evento experimental obteniéndose las gráficas de posición y velocidad angular de los eslabones del robot, ya que se desarrolló todo el sistema de adquisición de datos necesario. Los resultados fueron pobres debido a que el tren de engranaje del robot que se utilizó empezó a tener desgaste y flexión en los dientes de los engranes y en algunos intentos las bandas se aflojaban y se desajustaba la posición calculada de los eslabones. Para este tipo de experimentos se requiere cambiar los parámetros de diseño en el sistema de transmisión del robot y construirlo más robusto y con transmisión directa. Cabe mencionar, que en este evento la velocidad de las esferas se midió con un sistema de sensores infrarrojos que medía el tiempo que tardaban en pasar las esferas y así calcular su velocidad de salida sobre las ranuras.

Para la realización del segundo experimento se establecieron las dimensiones y la localización del sistema de visión y del robot industrial Mitsubishi<sup>®</sup>. Se realizaron las pruebas experimentales con el sistema de visión ofreciendo resultados satisfactorios y necesarios para este evento. Es importante resaltar, que como el sistema de visión era vulnerable a los cambios bruscos de luminosidad del medio ambiente, para solucionar el problema se dispuso de un cambio en la imagen, es decir, del espacio RGB al espacio HSV para procesar las imágenes capturadas y disminuir las posibles fallas debido a las variaciones de luz.

Con respecto a la integración de todo el sistema para el segundo experimento, es decir, del sistema de visión y el robot mitsubishi<sup>®</sup> se diseñó y se desarrolló un programa en Matlab<sup>®</sup> 7.0, ofreciendo resultados satisfactorios debido a que este programa es capaz de utilizar el puerto de comunicación RS232 para conectarse con el controlador del robot Mitsubishi<sup>®</sup> y el puerto USB para adquirir imágenes desde la cámara. Se programaron los valores de los ángulos que cada eslabón debía seguir para realizar la intercepción de las piezas. Se planteó la metodología para el evento experimental y se obtuvieron las funciones seguidoras y se analizó la trayectoria TPS y las funciones de las variables articulares en posición y velocidad. Lo anterior, conformó el montaje experimental que arrojó resultados aceptables.

Se resolvió el planteamiento del problema y se logró la intercepción de las piezas sin impacto. Una ventaja que se observó fue que la velocidad de la banda era baja, menos de medio metro por segundo y por ende, el tiempo que duró el evento experimental fue grande comparado con el tiempo que necesitó la computadora para la adquisición, el procesamiento de las imágenes y el cálculo de TPS. Lo anterior, permitió que los resultados fueran satisfactorios y realizables sin complicaciones mayores. Sin embargo,

es importante mencionar que la velocidad de la banda puede variar, disminuir o aumentarse y el sistema de visión puede responder dentro de un límite considerable de velocidad para lograr disminuir aun más los tiempos de alimentación de los módulos del sistema de manufactura flexible (FMS).

Cabe mencionar, que aunque las piezas no siempre estén dentro de la distancia máxima, para que el robot las capture, al menos con el hecho de que en un día de operación del sistema las piezas que pueda tomar el robot en forma múltiple representará un ahorro en el tiempo de alimentación de los módulos. Es decir, habrá momentos en que no pasen piezas cuya distancia entre ellas no esté dentro de la distancia máxima en la cual el robot pueda interceptarlas, pero las que se encuentren dentro de la distancia máxima serán capturadas.

Se realizaron una infinidad de eventos dentro de los cuales se obtuvieron resultados favorables y satisfactorios, es decir, de cada 10 intentos 6 fueron exitosos, los restantes fueron fallos debido al procesamiento irregular de la computadora.

Es importante mencionar que en este tipo de robot industriales no se pueden modificar las ganancias del controlador, no se pueden leer las señales de los encoders para realizar las gráficas de posición y velocidad, por tal motivo, solo se presentan las tablas con los valores leídos desde el buffer de las posiciones finales para cada punto alcanzadas por el efector final del robot.

## VI.18 REFERENCIAS

- [6.1] Otsu, N., "A Threshold Selection Method from Gray-Level Histograms," IEEE Transactions on Systems, Man, and Cybernetics, Vol. 9, No. 1, 1979, pp. 62-66.
- [6.2] Digital Image Processing, Rafael C. Gonzalez/ Richard E. Woods  
Second Edition, pag. 299, Prentice Hall 2002.
- [6.3] Manual de operación de robot industrial micro-robot system model: Movemaster EX RV-M1 Mitsubishi. número: BFP-A5191E-B
- [6.4] Manual de operación A465 Robot Arm USER GUIDE for use with C500C Controller, CRS Automation solutions, número: UMI-33-465-A



---

## CONCLUSIONES

En el presente trabajo de investigación se describe básicamente el desarrollo de una metodología propuesta que analiza y estudia la intercepción múltiple de objetos que describen trayectorias conocidas con aceleración constante. Se realizaron varias validaciones de los algoritmos con diferentes trayectorias de partículas libres, entre ellas el tiro parabólico con diferentes posiciones de disparo de las partículas y posiciones de intercepción diferentes. Además, se realizaron validaciones con intercepciones con impacto y sin impacto entre las partículas libres TPL1 y TPL2 y la trayectoria seguidora TPS.

Es importante hacer énfasis en la originalidad de este trabajo, se revisó la bibliografía con respecto a la intercepción múltiple de trayectorias y no hay trabajos hasta el momento donde los robots intercepten dos o más partículas en movimiento, lo anterior le da originalidad a este trabajo de investigación.

Los resultados se presentaron con simulaciones de algunos eventos de intercepción y dos aplicaciones experimentales para validar en forma teórica y práctica los algoritmos generados por dicha metodología, generando los resultados que a continuación se mencionan:

Las principales conclusiones de este trabajo son:

■ En la parte teórica:

- Para resolver el problema de recuperar los tiempos que se requieren para que el robot manipulador espere y tome una pieza que se mueve sobre una banda transportadora en un Sistema de Manufactura Flexible (FMS), se propuso una metodología de estudio que toma como datos la información (funciones) del movimiento de dichas piezas (posición, velocidad y distancia entre ellas) y generar un algoritmo en forma de función seguidora. Esta última se implementó al modelo cinemático del robot para que el efector final siga la trayectoria de dicha función e intercepte en forma múltiple dos piezas en pleno movimiento. Con lo anterior, se evita que el robot espere a que las piezas se detengan en la estación para que las tome.
- Se realizaron varias simulaciones para validar los algoritmos tomando como datos iniciales diferentes valores numéricos para mostrar los diferentes comportamientos que tenía la función seguidora para interceptar los objetivos móviles. Los resultados fueron satisfactorios, incluso se desarrolló un simulador para realizar esta tarea. Por otro lado, la robustez de los algoritmos en términos de la aceleración es buena, pueden seguir trayectorias cuya aceleración es constante pero también puede ser no constante.
- El modelo cinemático del robot de dos gdl y el del robot industrial Mitsubishi® se obtuvieron utilizando el álgebra de Cuaterniones. Las ecuaciones del modelo

cinemático del robot industrial se pudieron simplificar, reduciendo así, el número de operaciones y el tiempo de cómputo necesario para resolver la cinemática inversa en el tiempo de operación del sistema. Además, se planteó un algoritmo para encontrar el modelo cinemático para robots de  $n$  grados de libertad con juntas prismáticas o de revoluta (giratorias).

- En este trabajo se consideraron las dos técnicas de intercepción de trayectorias. La técnica “snatching” (arrebatarse) y la técnica llamada “catching” (atrapar). La técnica “snatching”, utiliza un tiempo más pequeño para interceptar la trayectoria, debido a que solo se anticipa al punto de intercepción sin importar la velocidad con que llega, esto implica un impacto o colisión con el objeto en movimiento.
- En la técnica “catching”, se requiere más tiempo para interceptar el objetivo, debido a que la trayectoria seguidora debe igualar la velocidad del objeto en algún instante de tiempo para atraparlo evitando la colisión o el impacto. La metodología propuesta funcionó para ambos casos y para validar los resultados se implementaron las funciones seguidoras en varias simulaciones y con dos experimentos, dando resultados aceptablemente satisfactorios.

■ En la parte experimental:

La función seguidora generada de la metodología de recuperación de tiempos, fue aplicada a un par de experimentos, primero a un prototipo de robot manipulador de dos grados de libertad y después al robot industrial Mitsubishi® que forma parte del Sistema de Manufactura Flexible (FMS), los resultados fueron los siguientes:

- En el prototipo de robot interceptor se utilizó una canasta para capturar los objetos en caída libre. Lo anterior, evitó diseñar y construir un efector final doble que quedaba fuera del alcance de este proyecto y permitió llevar a cabo el experimento, enfocándose solo a la validación del algoritmo en estudio. Además, se utilizó un sistema de control de movimiento bajo un esquema de control PID con prealimentación y retroalimentación de velocidad para que el efector final del prototipo alcanzara la referencia de posición y velocidad que demandaba la trayectoria seguidora TPS. Los resultados fueron pobres debido a que la estructura del robot no soportó las altas aceleraciones, las fricciones no modeladas y la vibración ocasionada por el tren de engranaje del prototipo que se utilizó para reducir la velocidad de los servomotores y adecuarlos a la tarea, esto ocasionó que el desempeño en el control en lazo cerrado durante el seguimiento de las trayectorias fuera empeorando. Sin embargo, se logró que de cada 10 intentos 6 fueran asertivos.
- Con respecto a la aplicación de los algoritmos en el robot industrial Mitsubishi®, la limitante en esencia se debió a la velocidad de cómputo para procesar las imágenes y calcular la velocidad y posición de las piezas, Sin embargo, se lograron resultados satisfactorios y de interés, ya que la parte de control y la

dinámica del robot están implícitos en el mismo, solo fue necesario establecer la comunicación entre el controlador del robot y el sistema de visión a través del ordenador que tenía el programa con el algoritmo programado.

- El sistema de visión desarrollado para llevar a cabo la parte experimental del robot industrial obtuvo la información cinemática de las piezas en movimiento, Sin embargo, es necesario señalar la limitante que se presentó cuando por los cambios bruscos de iluminación en algunos eventos el sistema se disparaba en ausencia de la piezas dentro del área de visión del sistema. La frecuencia de disparos en falso disminuyó considerablemente con el cambio de espacio RGB a el espacio HSV presentando un mejor desempeño.

Es importante, hacer énfasis en que se cumplieron los objetivos planteados en este trabajo, a pesar de las limitaciones antes mencionadas.

La metodología propuesta por ser novedosa y por su capacidad para generar funciones seguidoras que intercepten en forma múltiple trayectorias conocidas puede aplicarse para otro tipo de problemas, como por ejemplo; para realizar búsquedas de rescate de objetivos cuyo movimiento es conocido.

Por otro lado, en la industria el ahorro de tiempo significa un aumento en productividad y una mejor estructura para afrontar la competitividad mundial, es decir, la globalización. Por lo tanto, el problema de recuperar los tiempos muertos y disminuir los tiempos de fabricación al máximo ha llamado la atención y se ha atacado este problema con técnicas de diversa índole, desde la planeación de la producción hasta la modificación de la infraestructura interna de la empresa. Este trabajo, presenta una posible alternativa que utiliza los mismos recursos del Sistema de Manufactura Flexible (FMS) para lograr a través de la programación de los algoritmos de recuperación de tiempos, eficiencia en el proceso de paletización.

## TRABAJOS FUTUROS

En este tema se proponen algunos trabajos futuros partiendo del desarrollo de este trabajo. En realidad, son posibles mejoras que se fueron presentando a lo largo del desarrollo del proyecto.

1.- Se observó que la trayectoria TPS generada con la metodología propuesta, empezaba su movimiento con desplazamientos grandes comparados con los desplazamientos pequeños aproximadamente a la mitad de la trayectoria. Lo anterior implica que el robot en su inicio de movimiento utilice un voltaje y corriente grandes

para romper la inercia del robot y alcanzar la velocidad demandada. Este problema se puede solucionar aplicando algún algoritmo que distribuya uniformemente los desplazamientos de la función seguidora TPS, como por ejemplo el “espaciamiento de Chebychev” por nombrar alguno.

2.- Otro punto importante que se observó, es que fue necesario el diseño y construcción de un efector final doble acoplado al robot interceptor para capturar los objetos. En este trabajo, se utilizó una red y un gripper convencional, sin embargo, sería una excelente contribución la implementación de un doble efector final para llevar a cabo una amplia variedad de aplicaciones.

3.- Otro posible trabajo aparece cuando al capturar piezas de peso considerable, surge el problema de ajustar las ganancias del controlador del robot debido a que los parámetros del controlador cambiaron al capturar el primer objeto porque se añadió masa al sistema, Una forma de resolverlo sería aplicando algún esquema de control adaptable para ajustar las ganancias en el tiempo de operación del sistema y evitar que el robot pierda la trayectoria preestablecida para capturar el segundo objeto.

4.- Un problema adicional que se presentó con el sistema de visión fue su lentitud para realizar el procesamiento de imágenes y calcular la posición y velocidad de las piezas a interceptar. De lo anterior, se propone implementar los algoritmos desarrollados en este trabajo en algún sistema que procese con mayor rapidez la información, podría ser un DSP (Digital Signal Processor).

5.- En este trabajo, solo se capturaron dos piezas que se movían en línea recta una detrás de la otra sobre la banda transportadora. Sin embargo, algunos Sistemas de Manufactura Flexible (FMS) cuentan con bandas transportadoras que mueven varias piezas distribuidas sobre el área de la banda, La configuración cámara en mano permitiría al robot rastrear piezas que se mueven sobre la banda y generar la trayectoria adecuada. Por lo tanto, en lugar de utilizar la configuración de cámara fija como se utilizó en este trabajo, se puede implementar la configuración cámara en mano montada en el efector final del robot para resolver el problema.

6.- Finalmente, la metodología que se desarrolló en este trabajo utiliza la información de objetos que se mueven en trayectorias bidimensionales, una aportación importante sería adecuar la metodología para que genere la trayectoria seguidora para interceptar objetos que se mueven en el espacio tridimensional.

7.- En la metodología se plantearon problemas de intercepción de trayectorias de partículas libres que se movían con aceleración constante. Un posible trabajo futuro sería modificar la metodología para realizar el estudio de la intercepción de partículas que describen trayectorias cuya aceleración este en función de la posición, de la velocidad, etc. es decir, que ésta no sea constante.

**POR RAZONES DE ESPACIO SE PRESENTAN IMPRESOS  
SOLO FRAGMENTOS DE CODIGO DE LOS PROGRAMAS  
DESARROLLADOS EN ESTE TRABAJO**

**LOS PROGRAMAS COMPLETOS SE ENCUENTRAN  
DISPONIBLES EN EL CD QUE ACOMPAÑA ESTA TESIS  
IMPRESA**

**ADEMÁS SE HAN INCLUIDO LOS VIDEOS DE LOS EVENTOS  
EXPERIMENTALES**

## ANEXO A

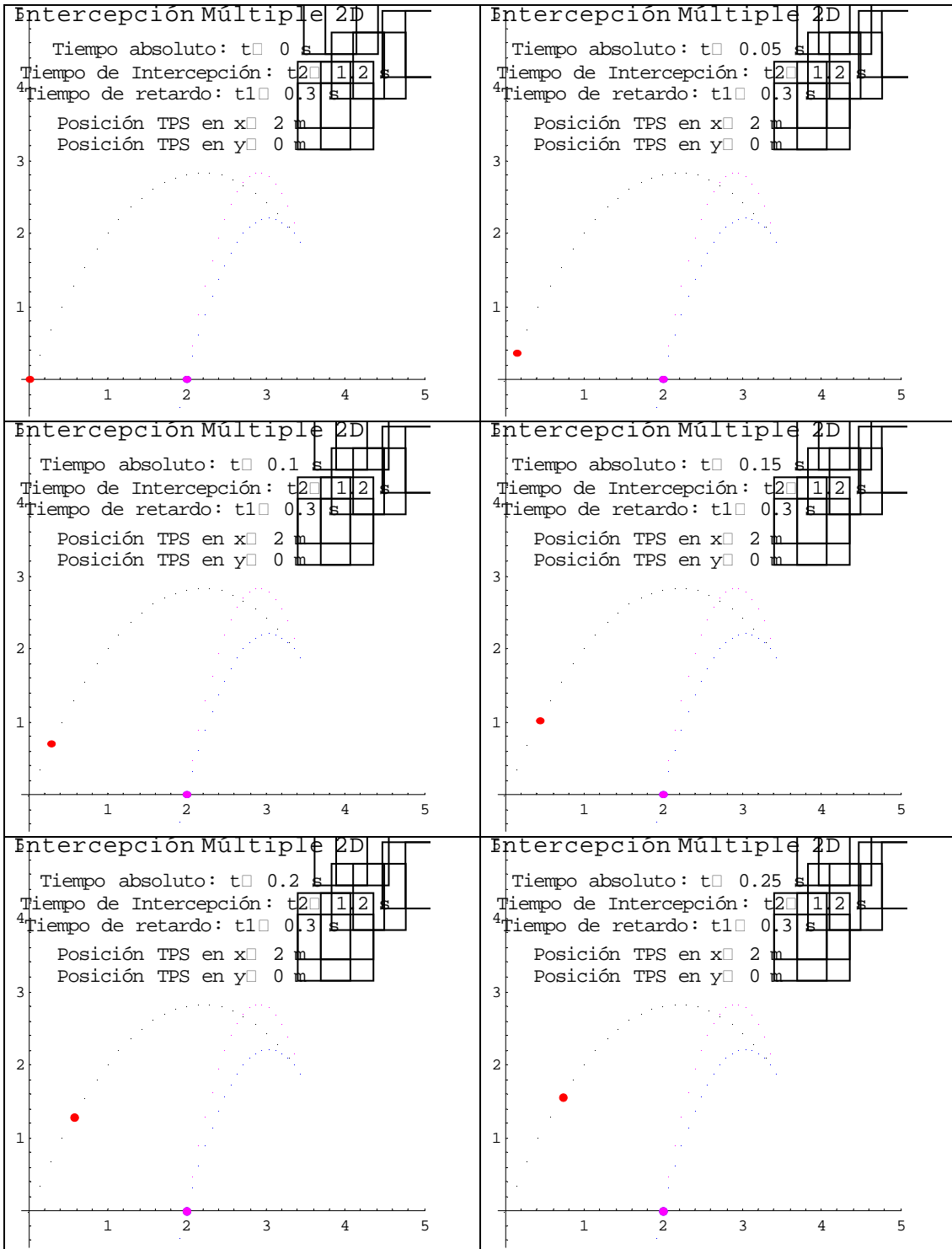
## Tiro\_par1cb.nb

Listado del programa que genera la intercepción de una partícula libre TPL con la trayectoria de la partícula seguidora TPS.

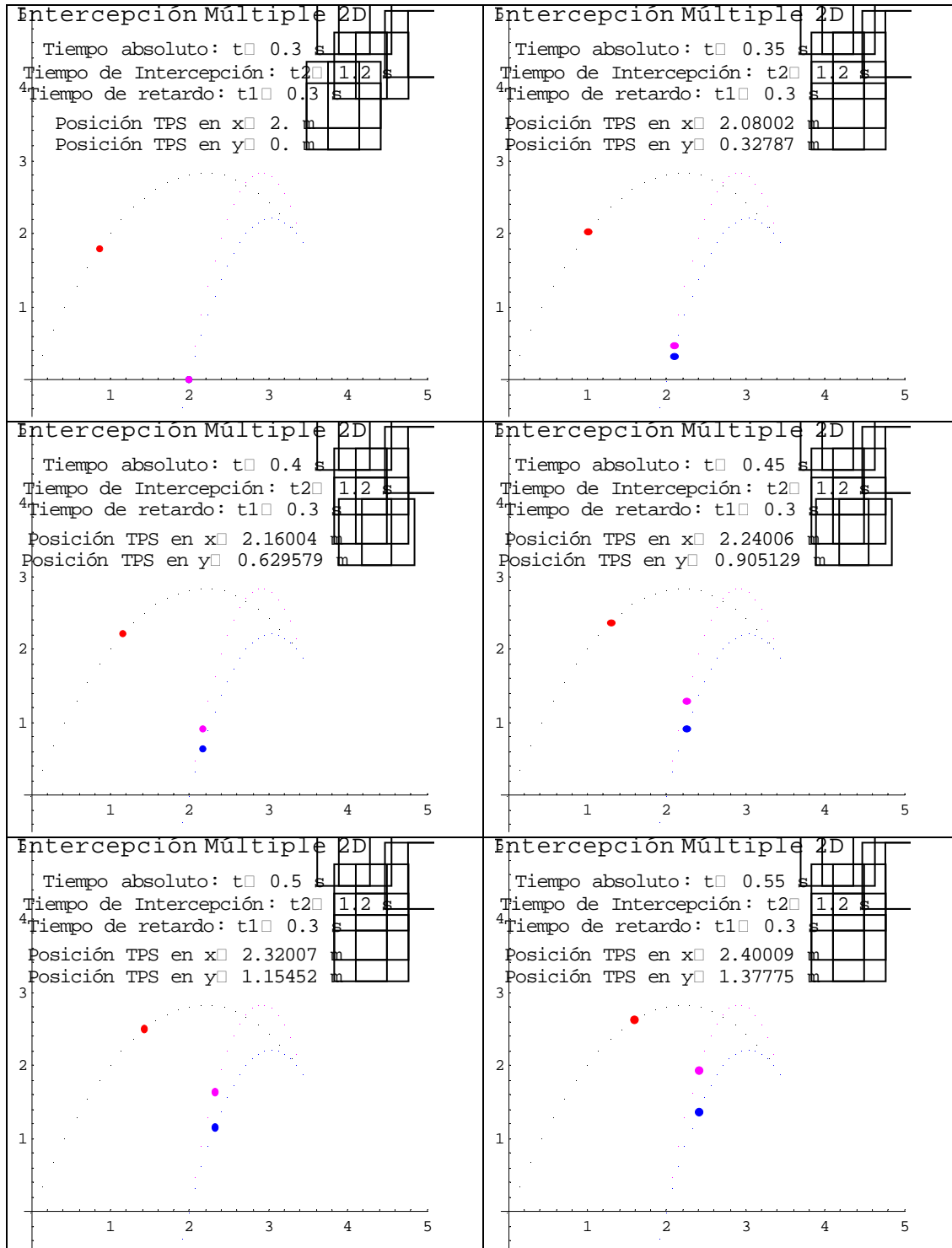
CODIGO EN MATHEMATICA ® 4.0	Descripción de comandos
<pre>ClearAll [t, t2, j1, t1, g, ang];</pre>	Limpia variables
<pre>V0 8;   69 Degree; g 9.81; t2 1.2; num t2 step; t1 0.3; step t2 24; a 2; b 0;</pre>	<p>Se definen los valores de la partícula libre.</p> <p>Tiempo de intercepción. Tiempo de la simulación total. Tiempo de retardo. Paso de integración 0.05 s.</p> <p>Coordenadas (<math>x^*(t)=a, y(t)=b</math>)</p>
<pre>level1   Text FontForm "Intercepción Múltiple 2D",     "Terminal", 16, 2, 5; level2   Text FontForm "Interceptado:",     "Terminal", 11, rr1 0.5, rr2 1;</pre>	<p>Leyendas que aparecen en la simulación.</p> <p>Esta leyenda aparece en el tiempo de intercepción <math>t_2</math>.</p>
<pre>p Table PointSize 0.007,   Point V0 Cos 0 t, V0 Sin 0 t - 0.5 g t^2,   t, 0, num step, step;</pre>	Se crea un vector con los puntos de la trayectoria de la partícula libre, con paso de integración igual a <b>step</b> .
<pre>caso1   Table PointSize 0.007, RGBColor 1, 0, 1,   Point c1 t2 V0 Cos 0 t1 a t2,     t2 t1,   c2 t2 V0 Sin 0 t1 t1,     t2 t1,   0.5 g t2^2 t1^2,   t, 0, num step, step;</pre>	Se crea un vector con los puntos de la trayectoria de la partícula seguidora para el caso I, con paso de integración igual a <b>step</b> .
<pre>caso2   Table PointSize 0.007, RGBColor 0, 0, 1,   Point rr1 t2 V0 Cos 0 t1 a t2,     t2 t1,   rr2 t2 V0 Sin 0 t1 t1,     t2 t1,   0.5 g t2^2 t1^2,   t, 0, num step, step;</pre>	Se crea un vector con los puntos de la trayectoria de la partícula seguidora para el caso II, con paso de integración igual a <b>step</b> .

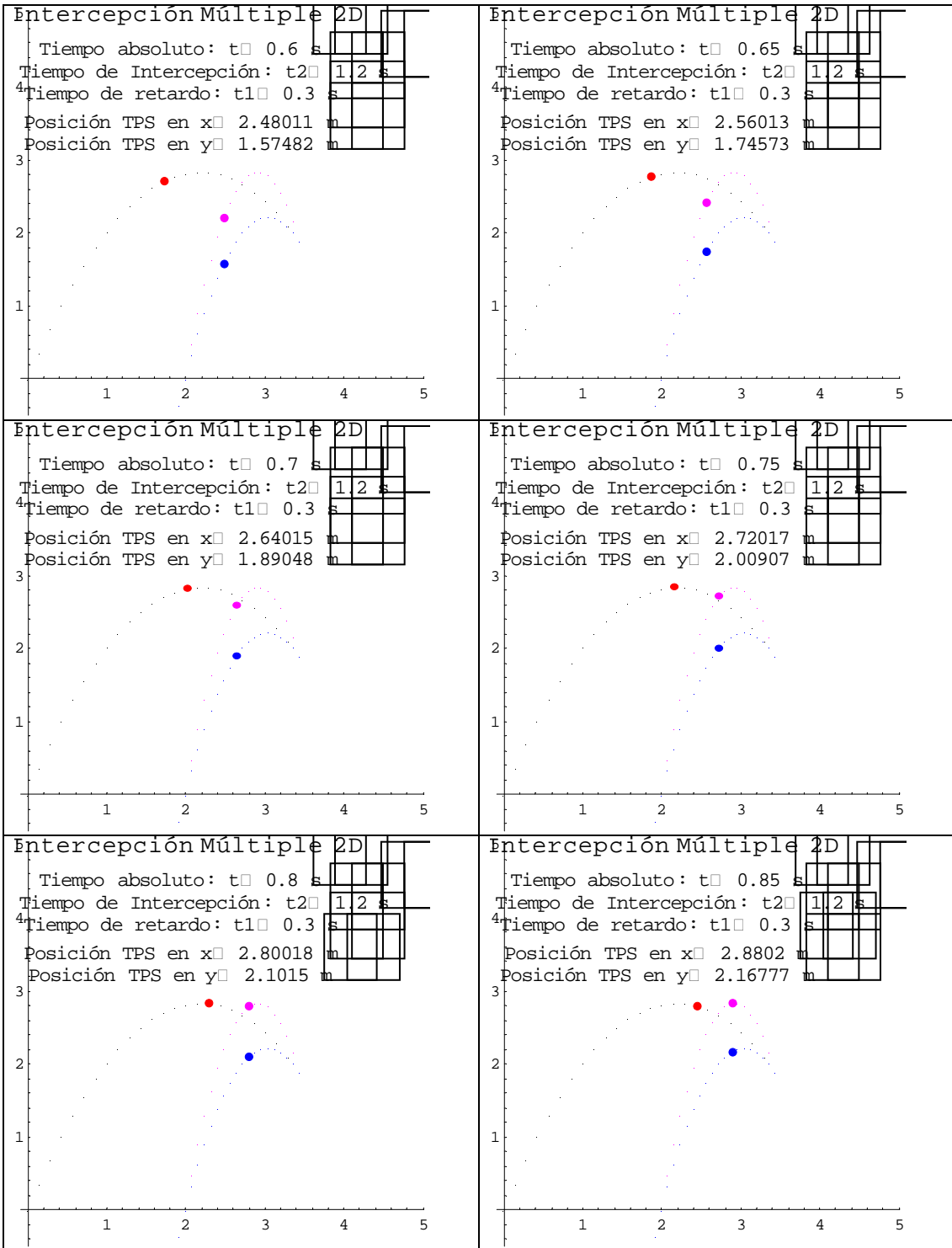
CODIGO EN MATHEMATICA ® 4.0	Descripción de comandos
<pre>rr1 a; rr2 0; c1 a; c2 0; For t 0, t mm, t step, Show Graphics {p, caso1, caso2, If t t2, lye2, lye1 t, Text FontForm "Tiempo absoluto: t " t s, "Courier", 12, 2, 4.5, Text FontForm "Tiempo de Intercepción: t2 " t2 s, "Courier", 12, 2, 4.2, Text FontForm "Tiempo de retardo: t1 " t1 s, "Courier", 12, 2, 3.9,</pre>	<p>Entra en un ciclo de repetición para graficar todos los instantes step y generar la secuencia de movimiento del fenómeno del tiro parabólico.</p> <p>Se incluye en la gráfica el texto donde se indica los valores de las posiciones de la partícula.</p>
<pre>r1 V0 Cos 0 t; r2 V0 Sin 0 t 0.5gt<sup>2</sup>;</pre>	<p>Ecuaciones paramétricas del tiro parabólico.</p>
<pre>rr1 a; rr2 0; c1 a; c2 0; If t t1, rr1 t2 V0 Cos 0 t H #1 a H t2 / t2 t1; rr2 t2 V0 Sin 0 t t1 t2 0.5gt<sup>2</sup> t2 t1 t2 t1<sup>2</sup> ; c1 t2 V0 Cos 0 t t1 a t t2 ; c2 t2 V0 Sin 0 t t1 t2 0.5gt<sup>2</sup> t2 t1 ; Null ;</pre>	<p>Condiciones iniciales en t=0 s.</p> <p>Condición de retardo. Cuando el tiempo absoluto es igual al tiempo de retardo <math>t=t_1</math> la partícula seguidora inicia su movimiento, para los casos I y II de intercepción.</p>
<pre>Text FontForm "Posición TPS en x" "rr1 m", "Courier", 12, 2, 3.5, Text FontForm "Posición TPS en y" "rr2 m", "Courier", 12, 2, 3.2, PointSize 0.02, RGBColor 1, 0, 0, Point t1, t2, PointSize 0.02, RGBColor 0, 0, 1, Point t1, rr2, PointSize 0.02, RGBColor 1, 0, 1, Point t1, c2</pre>	<p>Indica las posiciones de la trayectoria de la partícula seguidora.</p> <p>Gráfica de los puntos de la trayectoria, definiendo el tamaño del punto del y el color.</p>
<pre>PlotRange {0, 1, 5, 0, 5, 5}, AspectRatio 1, Axes True;</pre>	<p>Establece las propiedades de la gráfica. El tipo de ejes y la escala de ejes.</p>

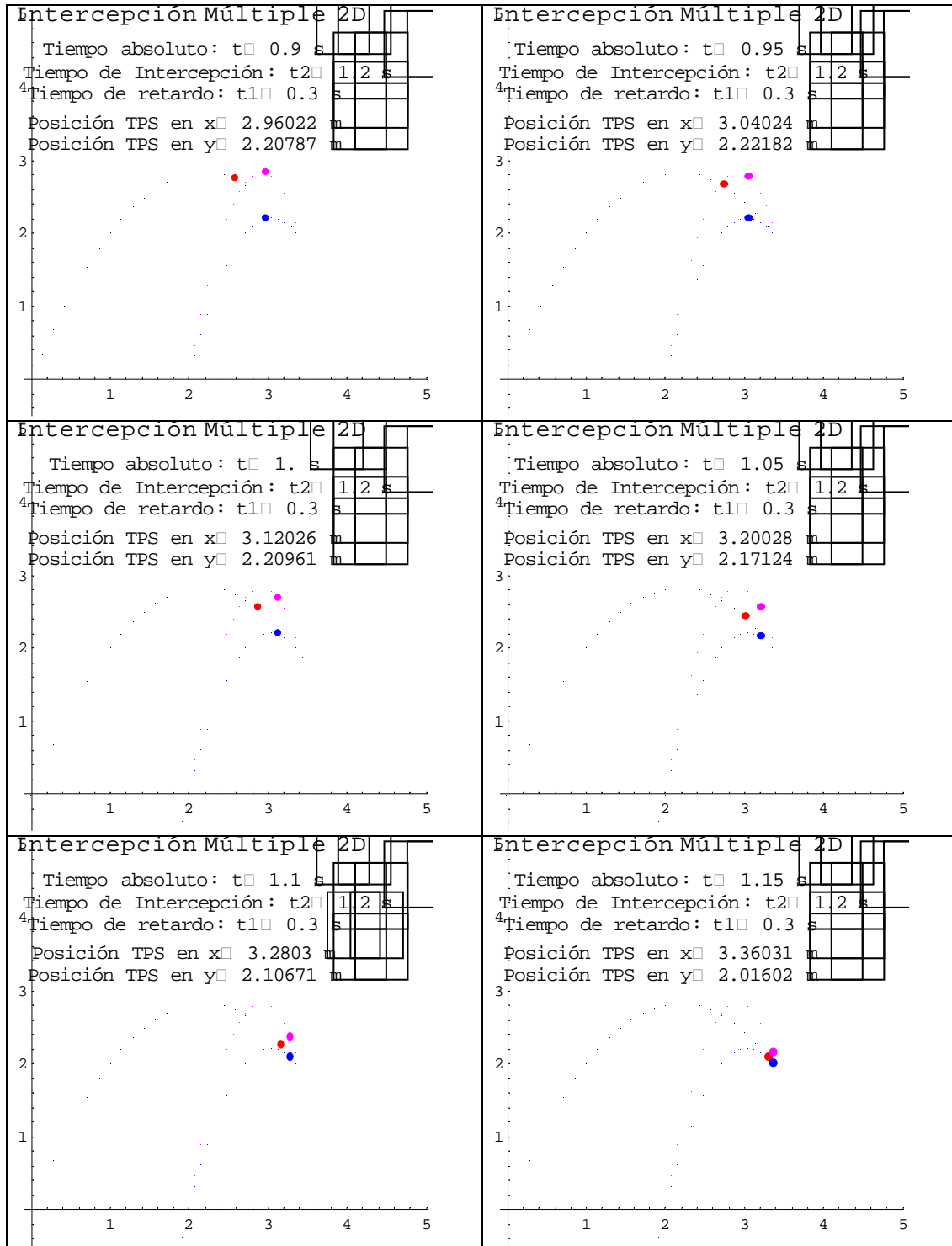
## Secuencia de gráficas generadas...

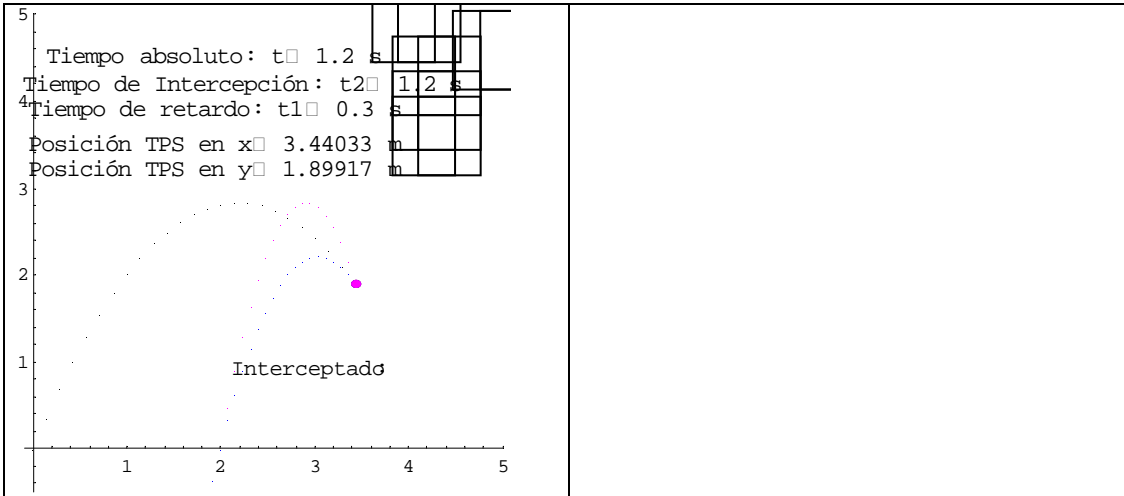










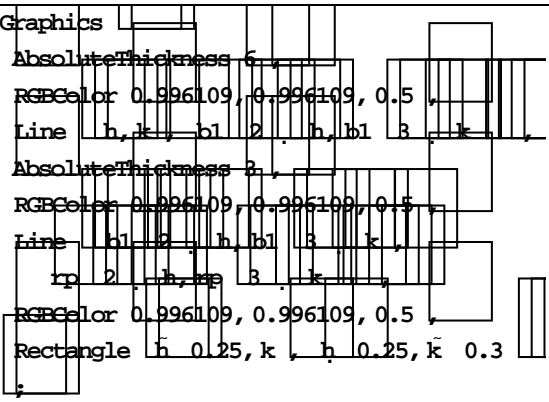


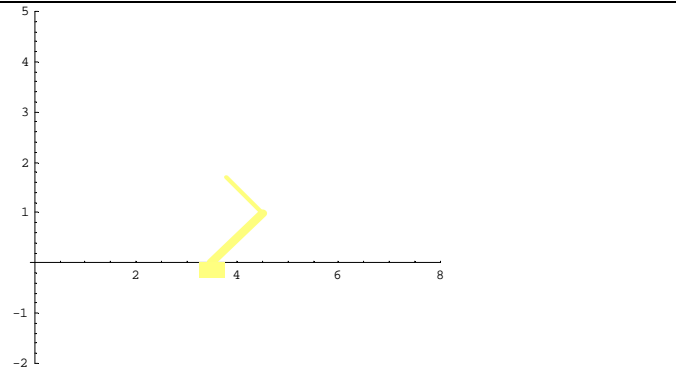
**Inte\_par1cb.nb**

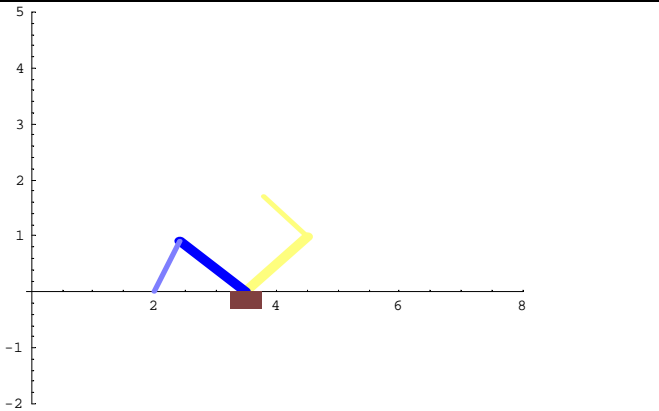
Listado del programa que genera la intercepción de una partícula libre TPL con el efector final de un robot manipulador de 2 gdl. La cinemática inversa se aplicó al caso I.

CODIGO EN MATHEMATICA® 4.0	Descripción de comandos
<pre> QuaterM P, Q := P 1 Q 1 P 2 Q 2 P 3 Q 3 P 4 Q 4 P 2 Q 1 P 1 Q 2 P 4 Q 2 P 3 Q 4 P 3 Q 1 P 4 Q 2 P 1 Q 2 P 2 Q 4 P 4 Q 1 P 2 Q 2 P 2 Q 2 P 1 Q 4 QuaterCon P := P 1 P 2 P 3 P 4 Ropq P, Q := QuaterM P, QuaterM Q, QuaterCon P := </pre>	<p>Operación de multiplicación de quaterniones.</p> <p>Conjugado de P.</p> <p>Rotación de Q con el parámetro P.</p>
<p><b>Configuración NO deformada.</b></p>	<p>Leyenda de configuración no deformada del robot manipulador.</p>
<pre>ClearAll p, q, rp, b1, b2, g1, p0, p1, q0, q1, h, k</pre>	<p>Limpia de las variables a utilizar.</p>
<pre>e1 := {0, 1, 0, 0}</pre>	<p>Base canónica de la dirección del eje x.</p>

$\alpha_1$ 45 Degree; $\alpha_2$ 90 Degree;	Ángulos iniciales de los eslabones del robot manipulador.
--	---

CODIGO EN MATHEMATICA <sup>®</sup> 4.0	Descripción de comandos
$d1$ 1.4; $d2$ 1;	Longitudes de los eslabones del robot manipulador de 2 gdl.
$p$ $\{p0, 0, 0, p3\}$ ; $q$ $\{q0, 0, 0, q3\}$ ;	Parámetros de los cuaterniones $p$ y $q$ .
$p0$ Cos $\frac{\alpha_1}{2}$ ; $p3$ Sin $\frac{\alpha_1}{2}$ ; $q0$ Cos $\frac{\alpha_2}{2}$ ; $q3$ Sin $\frac{\alpha_2}{2}$ ;	Cantidades de rotación para los cuaterniones $p$ y $q$ . Y eje de giro, en este caso será el eje Z.
$e11$ Ropq $p, e1$ ; $e12$ Ropq $q, e11$ ;	Aplicación de la transformación lineal de rotación. Para los dos eslabones,
$b1$ $d1 e11$ ; $b2$ $d2 e12$ ;	Representación de los eslabones 1 y 2 a través de los vectores $b1$ y $b2$ respectivamente.
$rp$ Simplify $b1 b2$ N	Vector de posición referido a la base global fija.
$h$ 3.5; $k$ 0;	Posición de la base del robot manipulador en el sistema de coordenadas fijo.
$h, 0$ 282843, 1.69706, 0.	
$g1$ Graphics 	Generación de las gráficas que despliegan el movimiento del robot manipulador de 2 gdl en su movimiento de intercepción de la trayectoria de la partícula libre caso I.
Show $g1$ , Axes True, PlotRange $\{0, 1\}, \{0, 8\}$ ; AspectRatio 1 ; Print "x ", $rp$ 2 $h$ N ; Print "y ", $rp$ 3 $k$ N ;	Muestra las gráficas generadas con el comando <i>Graphics</i> , y le asigna atributos a los ejes y escala.

	<p>La gráfica generada de la configuración no deformada del robot manipulador (posición de Home). Desde esta posición se tomará la referencia de los desplazamientos angulares de los eslabones 1 y 2.</p> <p>Desde el origen saldrán las partículas libres.</p>
<p>x <input type="checkbox"/> 3.78284 y <input type="checkbox"/> 1.69706</p>	<p>Posición del efector final</p>
<p><b>Configuración NO deformada para inicio de intercepción</b></p>	<p>Comentario</p>
<pre> ClearAll[R0, R3, S0, S3, a11, a12, b1, b2]; R0 = R0, 0, 0, R3; S0 = S0, 0, 0, S3; a11 = Ropq[R, e11]; a12 = Ropq[S, e12]; b1 = d1 a11; b2 = d2 a12; rp1 = Simplify[b1, b2, N]; </pre>	<p>Limpia variables a utilizar. Define los parámetros de los Cuaterniones R y S. Genera las rotaciones de los eslabones 1 y 2. Define los vectores que representan los eslabones 1 y 2. Genera la ecuación de posición de los eslabones para ser resuelta en la cinemática inversa del robot manipulador de 2 gdl.</p>
<pre> f1 = rp1[2][[1]] h; f2 = rp1[3][[1]] k; f3 = R0^2 - R3^2 - 1; f4 = S0^2 - S3^2 - 1; </pre>	<p>Se construye el sistema de ecuaciones no lineal del tipo polinomial de 4x4.</p>
<pre> sol = FindRoot[{f1, f2, f3, f4}, {R0, 0.1, R3, 0.6, S0, 0.1, S3, 0.6},   MaxIterations -&gt; 15]; resp = {R0, R3, S0, S3} /. sol; R0 = resp[1]; R3 = resp[2]; S0 = resp[3]; S3 = resp[4]; </pre>	<p>Se utiliza el comando de <i>FindRoot</i> para encontrar las soluciones numéricas. Con los valores iniciales mostrados para las variables. Ya que este comando está basado en el método de Newton-Rhapson y una variante del método de la secante para encontrar la solución con un máximo de 15 iteraciones. Guarda la solución en el vector <b>resp</b>. Asigna las soluciones a las variables R0, R3, S0, S3 para resolver la ecuación de posición <b>rp1</b>.</p>

<pre> g11 Graphics AbsoluteThickness 6 RGBColor 0, 0, 1 Line h, k, b1 2, h, b1 3, k AbsoluteThickness 2 RGBColor 0.5, 0.5, 0.994199 Line h1 2, h, h1 3, k rp1 2, h, rp1 3, k RGBColor 0.502, 0.251, 0.251 Rectangle h 0.25, k, h 0.25, k 0.3 </pre>	<p>Genera las gráficas del movimiento del robot para alcanzarla posición inicial de intercepción.</p> <p>Las propiedades de las graficas son el ancho de línea, el color de las líneas y las posiciones.</p>
<pre> Show g1, g11 Axes True PlotRange 0, 1, 8, 2, 5 AspectRatio 1 Print "x", rp1 2, h, N Print "y", rp1 3, k, N Print "1", Chop 2ArcSin R3 Degree Print "2", Chop 2ArcSin S3 Degree 2ArcSin R3 Degree </pre>	<p>Muestra la secuencia de graficas que darán la animación del movimiento del robot.</p> <p>Despliega los valores de los ángulos en grados y las dos soluciones. Ya que el sistema es cuadrático.</p>
	<p>Esta es la gráfica que se despliega de la posición inicial desde la cual el robot interceptará la trayectoria de la partícula libre. El efector final se encuentra en la posición (2,0) referido al sistema global fijo.</p> <p>Los cálculos anteriores permitieron determinar la cantidad de rotación de cada eslabón para color el efector final en la posición deseada (2,0).</p>
<pre> x 2. y 1.06568 10^7 1 94.8435 2 14.689 </pre>	<p>Posición del efector final y desplazamientos angulares de los eslabones 1 y 2, respectivamente.</p>
<p><b>Acoplamiento Robot particulas seguidoras movimiento relativo</b></p>	<p>Leyenda que indica el acoplamiento y la sincronización de los movimientos del robot con la trayectoria libre.</p>

<pre> ClearAll t, t2, t3, t1, g; V0 = 8; alpha = 69 Degree; g = 9.81; t2 = 1.2; num = t2 / step; t1 = 0.3; step = t2 / 24; a = 2; </pre>	<p>Limpia las variables a utilizar. Se definen los valores y los datos para las ecuaciones paramétricas del tiro parabólico. Se establece el tiempo de retardo <math>t_1</math>. El tiempo de intercepción <math>t_2</math> y el desplazamiento a unidades sobre el eje X del cambio de base.</p>
<pre> level1 = Text FontForm "Intercepción Múltiple 2D", "Terminal", 16, 2.6, 4.9; level2 = Text FontForm "Interceptado:", "Terminal", 11, rr1 - 0.5, rr2 - 1; </pre>	<p>Despliega de leyendas con información que indica cuando ocurrió la intercepción.</p>
<pre> p = Table PointSize 0.007, Point V0 Cos alpha t, V0 Sin alpha t - 0.5 g t^2, t, 0, num / step, step; </pre>	<p>Se genera un vector de los puntos del tiro parabólico con los datos antes capturados.</p>

<pre> caso1 = Table PointSize 0.007, RGBColor 1, 0, 1, Point c1 t2 V0 Cos alpha t1 - a t t2, t2 t1, c2 t2 V0 Sin alpha t1 - t1 t1, t2 t1, 0.5 g t^2 t t1^2, t2 t1^2, t, 0, num / step, step; </pre>	<p>Se genera un vector con los puntos de la trayectoria de la partícula seguidora caso II. Además, se define el color y el tamaño de punto.</p>
<pre> caso2 = Table PointSize 0.007, RGBColor 0, 0, 1, Point rr1 t2 V0 Cos alpha t1 - a t t2, t2 t1, rr2 t2 V0 Sin alpha t1 - t1 t1, t2 t1, 0.5 g t^2 t t1^2, t2^2 t1^2, t, 0, num / step, step; </pre>	<p>Se genera un vector con los puntos de la trayectoria de la partícula seguidora caso I. Además, se define el color y el tamaño del punto.</p>
<pre> rr1 = a; rr2 = 0; c1 = a; c2 = 0; </pre>	<p>Condiciones Iniciales.</p>



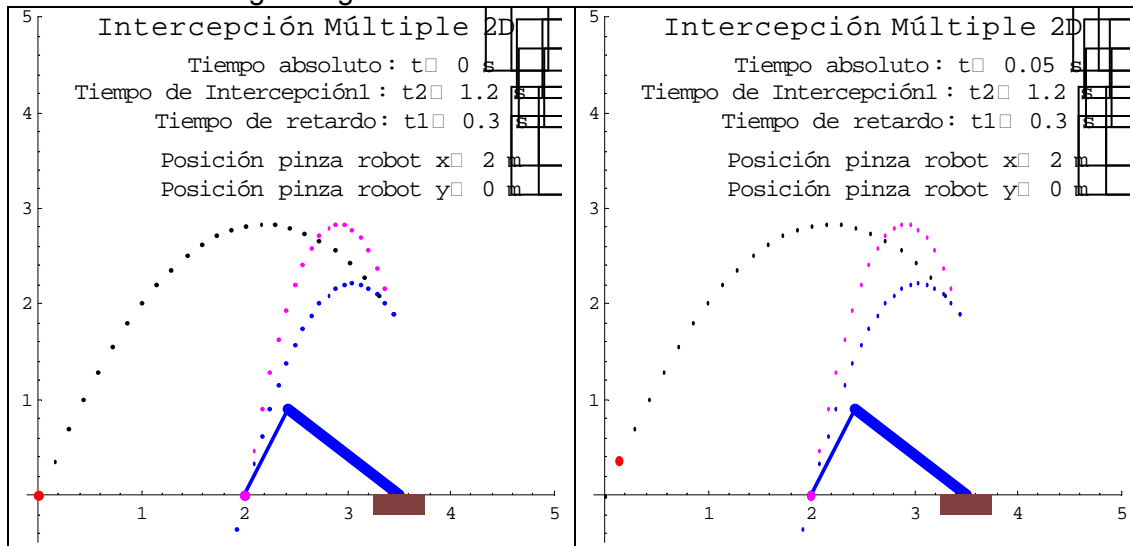
<pre> i 0; For t 0, t num, t step, g2 Graphics p, caso1, caso2, If t t2, ley2, ley1 Text FontForm "Tiempo absoluto: t " t s, "Courier", 12, 3, 4.5 Text FontForm "Tiempo de Intercepción: t2 t2 s, "Courier", 12, 2.6, 4.2 Text FontForm "Tiempo de retardo: t1 " t1 s, "Courier", 12, 3, 3.9 </pre>	<p>Se crea un ciclo iterativo para graficar cada una de las posiciones de las partículas libre y seguidora que en este caso sigue el efector final del robot interceptor.</p> <p>Se crean los textos que van informando de las posiciones y tiempos del desarrollo del evento.</p>
<pre> r1 V0 Cos 0 t; r2 V0 Sin 0 t 0.5gt^2; r11 a; r22 0; c1 a; c2 0; </pre>	<p>Se definen las ecuaciones del tiro parabólico.</p> <p>Se definen las condiciones iniciales.</p>
<pre> If t t1, rr1 t2 V0 Cos 0 t2 t1 a t2 t1; rr2 t2 V0 Sin 0 t2 t1 0.5gt2^2 t2 t1^2; c1 t2 V0 Cos 0 t2 t1 a t2 t1; c2 t2 V0 Sin 0 t2 t1 0.5gt2^2 t2 t1^2; Null ; </pre>	<p>Se crea una condición para cuando el tiempo absoluto es igual al tiempo de intercepción, para los casos I y II</p>
<pre> Text FontForm "Posición pinza robot x" "rr1 m, "Courier", 12, 3, 3.5 Text FontForm "Posición pinza robot y" " rr2 a, "Courier", 12, 3, 3.2 PointSize 0.02, RGBColor 1, 0, 0 Point =r1, r2 PointSize 0.02, RGBColor 0, 0, 1 Point =c1, c2 PointSize 0.02, RGBColor 1, 0, 1 Point c1, c2 ; </pre>	<p>Despliega el texto que contiene la información del evento. Posición de la pinza. Localización de los puntos que generan la trayectoria referido al sistema de coordenadas fijo y el tiempo transcurrido.</p>
<p><b>CINEMÁTICA INVERSA PARA EL ROBOT DE 2 GDL</b></p>	<p>Comentario dentro del programa que indica el inicio del desarrollo del código para la cinemática Inversa del robot interceptor</p>

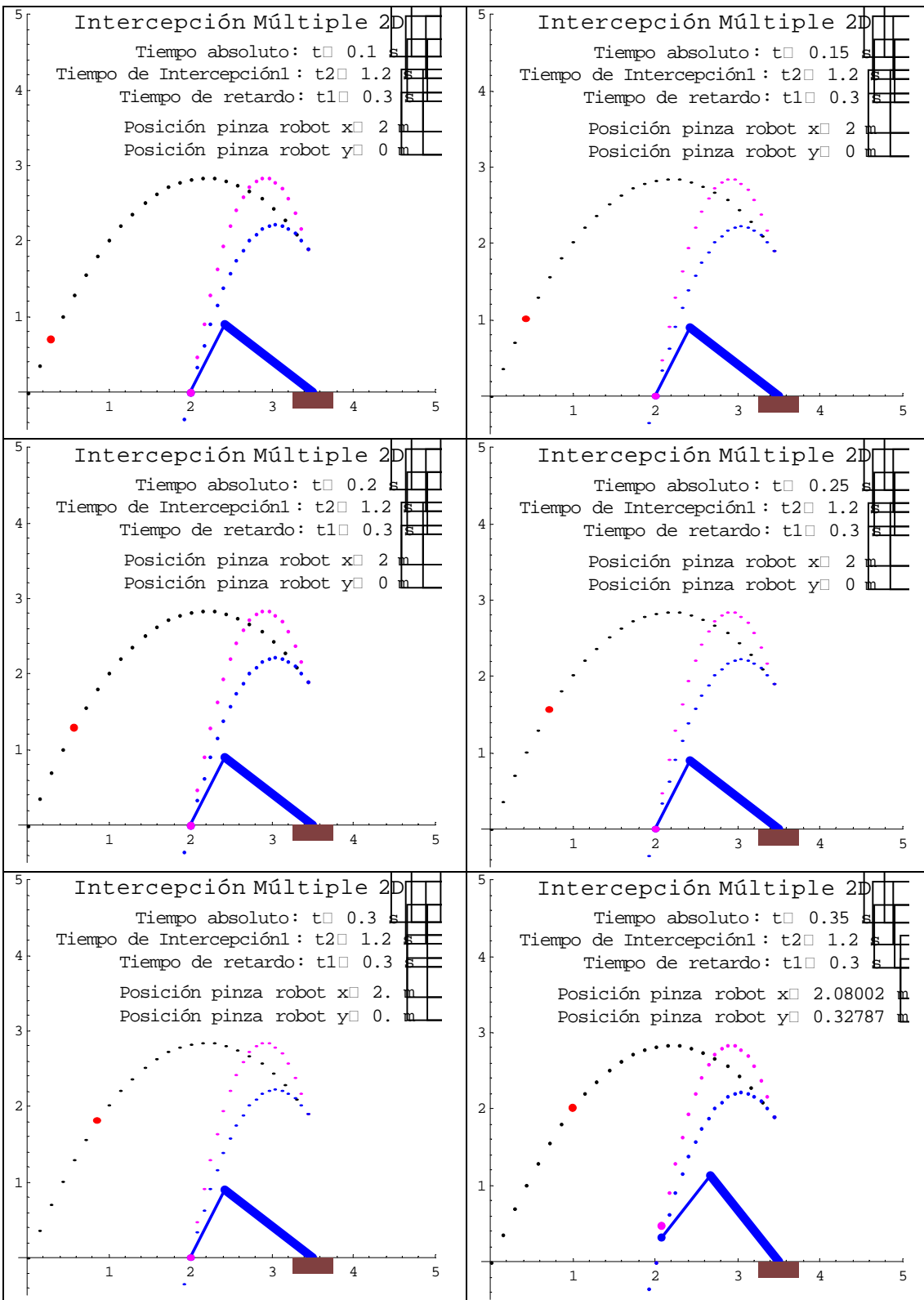
<pre>ClearAll R, S, R0, R3, S0, S3, b1, b2, k1, k2, sol, f1, f2, f3, f4 ;</pre>	Limpia variables a utilizar.
<pre>R[] R0, 0, 0, R3 ; S[] S0, 0, 0, S3 ; k1[] Ropq R, e1 ; k2[] Ropq S, e2 ; b1[] d1 k1 ; b2[] d2 k2 ; rp2[] Simplify b1 b2 N ;</pre>	<p>Construye los Cuaterniones.          Genera las rotaciones de los eslabones 1 y 2.          Define los vectores de los eslabones 1 y 2.          Vector posición referido al sistema global fijo.</p>
<pre>x[] rr1 h ; y[] rr2 k ;</pre>	Se define el cambio de base para la posición del robot manipulador y para implementar el sistema de ecuaciones.
<pre>f1[] rp2 2 x ; f2[] rp2 3 y ; f3[] R0^2 R3^2 1 ; f4[] S0^2 S3^2 1 ;  iR0[] 0.7 ; iR3[] 0.1 ; iS0[] 0.7 ; iS3[] 0.1 ;</pre>	<p>Establece el sistema de ecuaciones a resolver utilizando un método numérico de aproximación.</p> <p>Se definen los valores iniciados de iteración.</p>

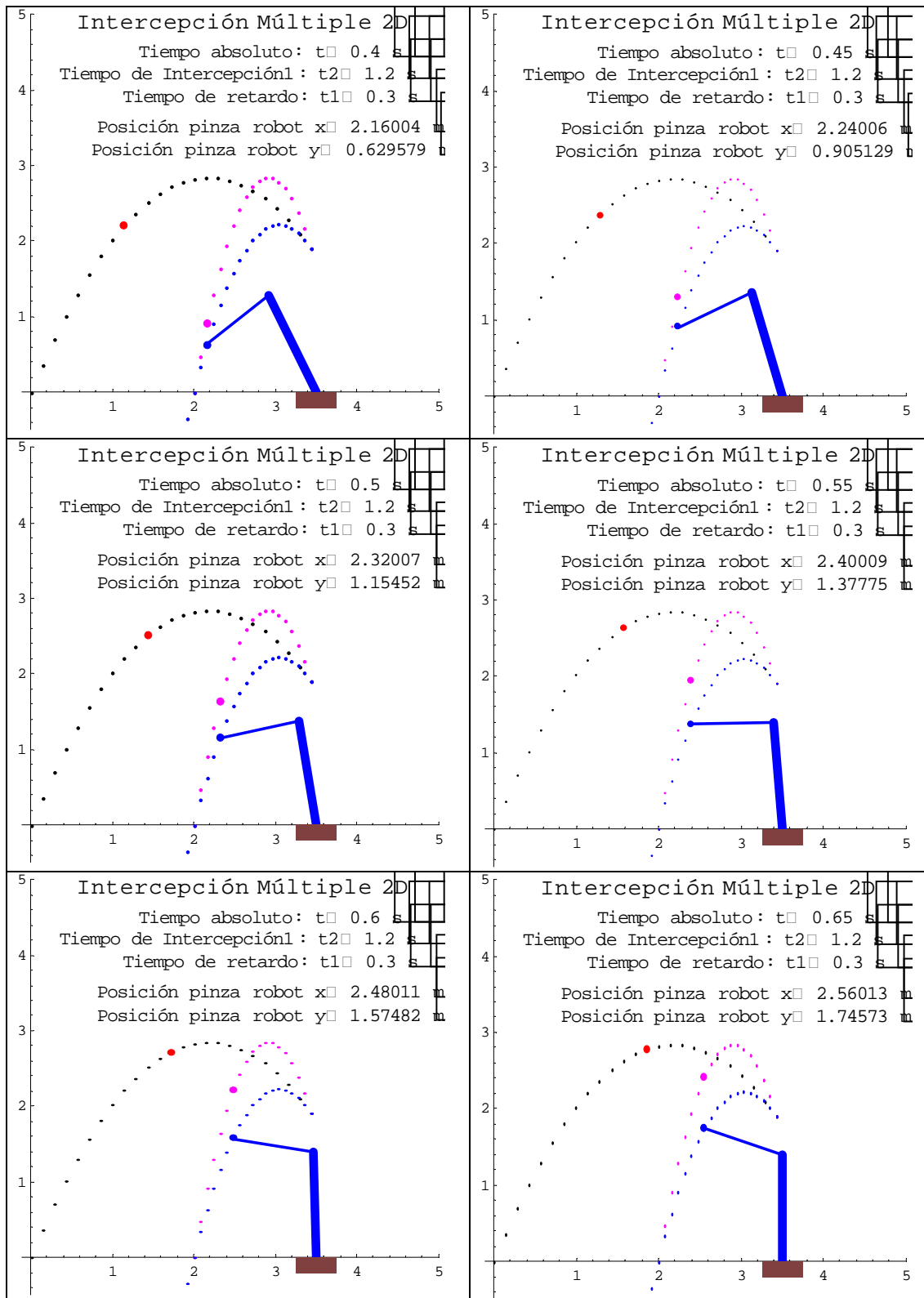
<pre>sol[] FindRoot {f1, f2, f3, f4, R0, iR0, R3, iR3, S0, iS0, S3, iS3, MaxIterations[] 15 ;</pre>	Se aplica el método numérico. Newton-Rhapon para sistemas de ecuaciones no lineales del tipo polinomial.
<pre>e5[] 10^-5 ; Presición requerida ; If Abs x - rp2 2 sol [] e5 &amp;&amp; Abs y - rp2 3 sol [] e5, Null, Print "Error" ;</pre>	Debido a que la solución numérica es una aproximación a la solución exacta, se calcula la precisión deseada e.
<pre>resp[] R0, R3, S0, S3 sol ; R0[] resp 1 ; R3[] resp 2 ; S0[] resp 3 ; S3[] resp 4 ;</pre>	Se guardan y se procesan las soluciones encontradas para después encontrar los desplazamientos angulares de los eslabones 1 y 2 del robot interceptor.

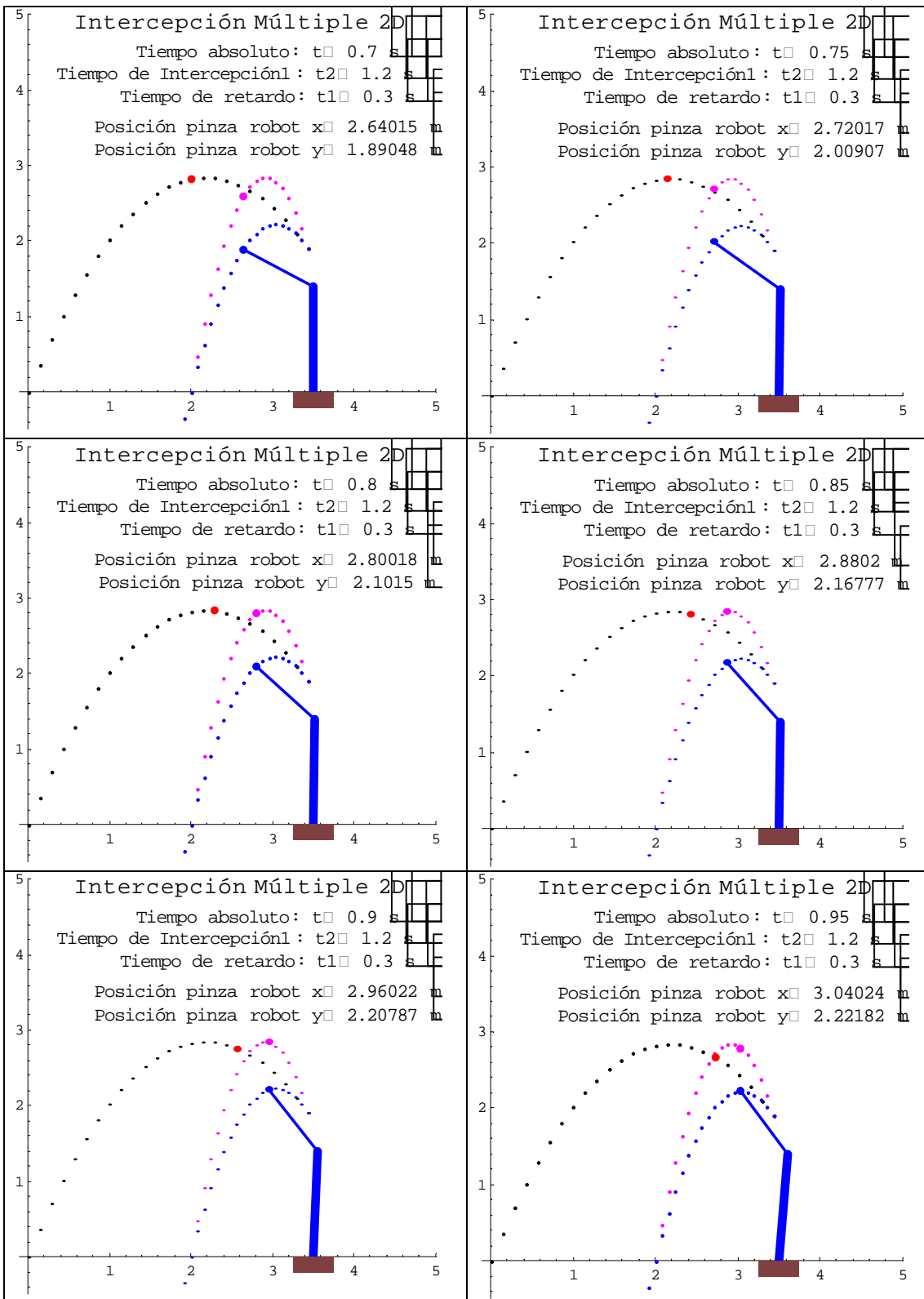
<pre> g12 Graphics AbsoluteThickness 6 RGBColor 0, 0, 1 Line h, k, h1, k1 AbsoluteThickness 2 RGBColor 0, 0, 1 Line k1, h, h1, k rp2 2, h, rp2 3, k, h RGBColor 0.502, 0.251, 0.251 Rectangle h, 0.25, k, h, 0.25, k, 0.2 </pre>	<p>Se generan las gráficas del movimiento del robot y del desplazamiento de la partícula seguidora en forma sincronizada. Es decir, la base del tiempo es la misma para los dos sistemas.</p>
<p>Fin de la CINEMÁTICA INVERSA.</p>	<p>Comentario.</p>
<pre> Show g12, g2, Axes True, PlotRange {0, 1, 5}, {0, 5, 5} AspectRatio 1 1; </pre>	<p>Despliega todas las gráficas generadas y muestra la animación de la secuencia de gráficas.</p>

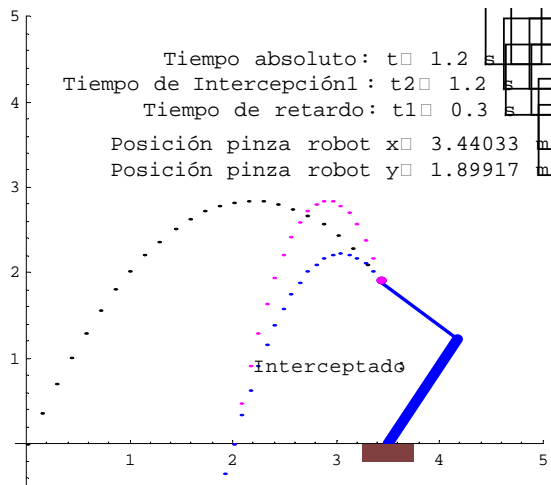
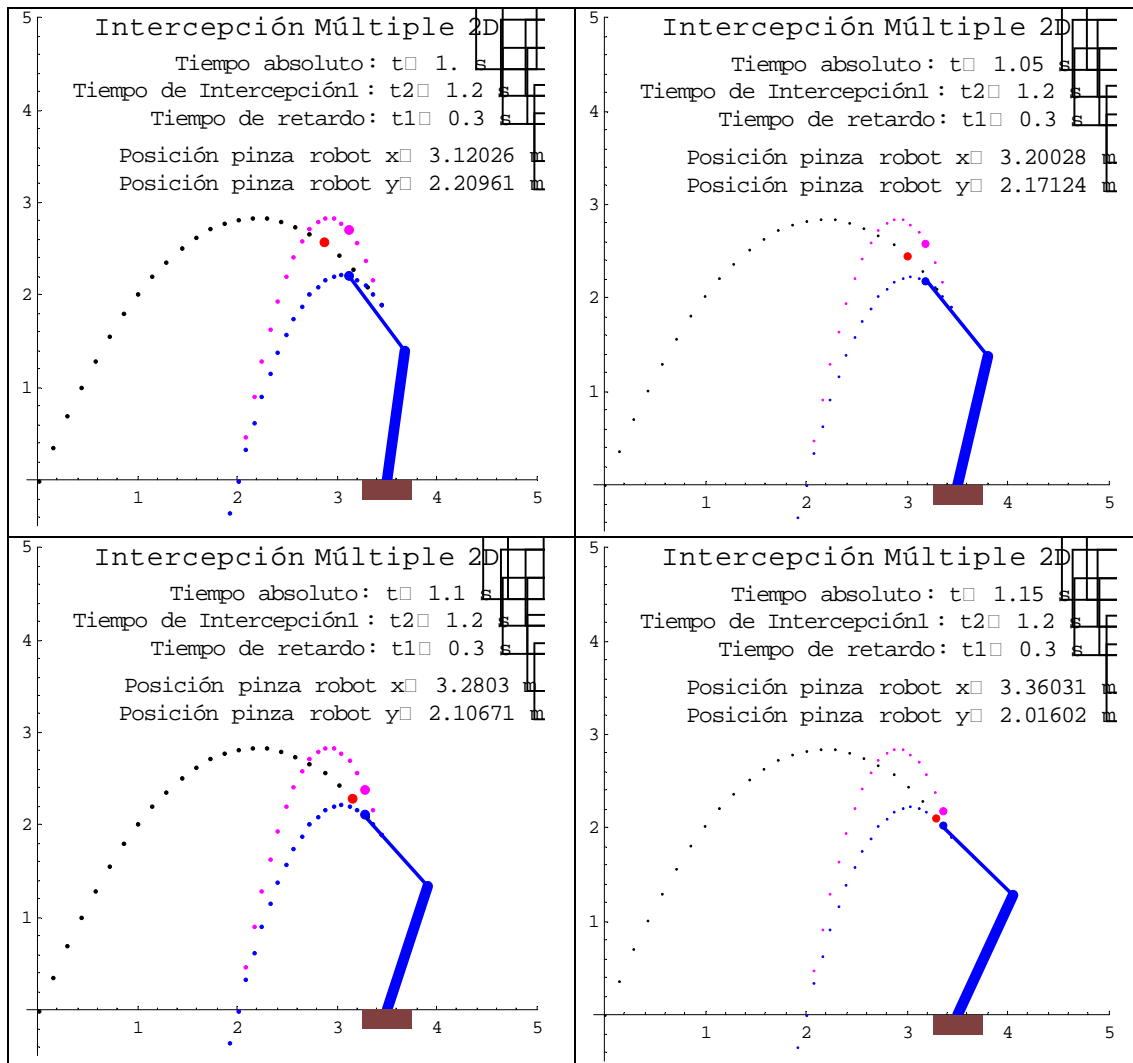
Secuencia de imágenes generadas.





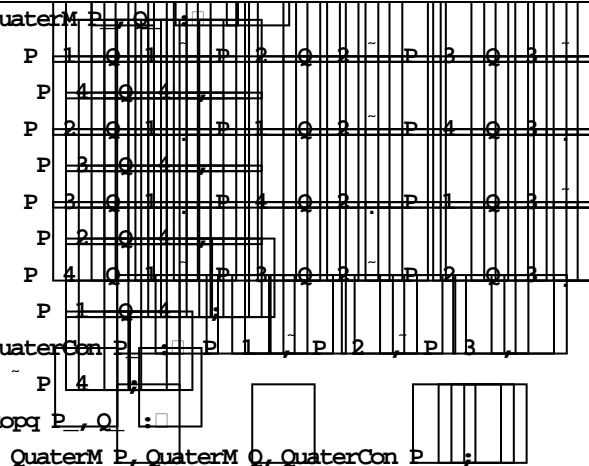
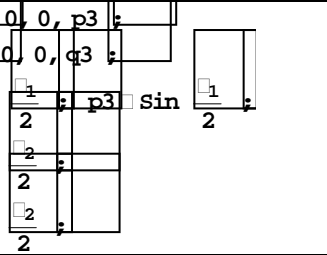
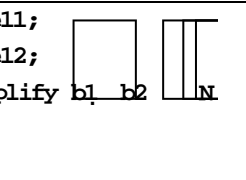




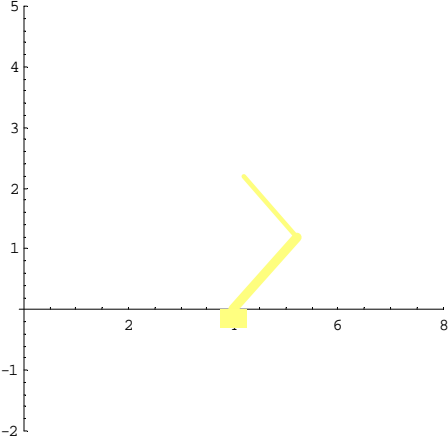


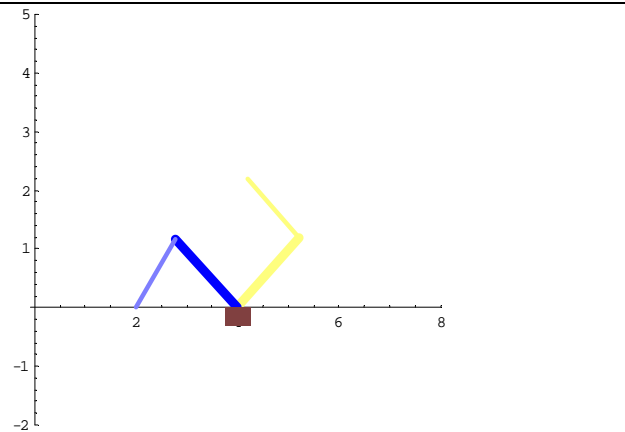
## Inte\_par2cb.nb

Listado del programa que genera la intercepción de dos partículas libres TPL1 y TPL2 con el efector final de un robot manipulador de 2 gdl. La cinemática inversa se aplicó al caso I. La explicación coincide con el listado de **Tiro\_par1cb.nb**.

CODIGO EN MATHEMATICA ® 4.0	Descripción de comandos
<pre>QuaterM P, Q, P 1 Q 1 P 2 Q 2 P 2 Q 2 P 4 Q 1 P 2 Q 1 P 1 Q 2 P 4 Q 2 P 2 Q 4 P 3 Q 1 P 4 Q 2 P 1 Q 2 P 2 Q 4 P 4 Q 1 P 3 Q 2 P 2 Q 2 P 1 Q 4 QuaterCon P, Q, P 1, P 2, P 3, P 4 Ropq P, Q QuaterM P, QuaterM Q, QuaterCon P</pre> 	Definición de la operación multiplicación
<b>Configuración NO deformada.</b>	Leyenda
<pre>ClearAll p, q, mp, b1, b2, g1, p0, p1, q0, q1, h, k e1 0, 1, 0, 0 i 45 Degree; j 90 Degree; d1 1.7; d2 1.4;</pre>	
<pre>p p0, 0, 0, p3 q q0, 0, 0, q3 p0 Cos <math>\frac{i_1}{2}</math> p3 Sin <math>\frac{i_1}{2}</math> q0 Cos <math>\frac{j_2}{2}</math> q3 Sin <math>\frac{j_2}{2}</math></pre> 	
<pre>e11 Ropq p, e1 e12 Ropq q, e11</pre>	
<pre>b1 d1 e11; b2 d2 e12; rp Simplify b1 b2 N h 4; k 0;</pre> 	
<pre>b, 0 212132, 2.19203, 0. 1</pre>	



<pre> g1 <input type="checkbox"/> Graphics AbsoluteThickness 6; RGBColor 0.996109, 0.996109, 0.5; Line h, k, b1 2, h, b1 3, k; AbsoluteThickness 2; RGBColor 0.996109, 0.996109, 0.5; Line b1 3, h, b1 3, k; rp 2, h, rp 3, k; RGBColor 0.996109, 0.996109, 0.5; Rectangle h 0.25, k, h 0.25, k 0.3; </pre>	
<pre> Show g1, Axes <input type="checkbox"/> True; PlotRange <input type="checkbox"/> {0, 1, 8, 2, 5}; AspectRatio <input type="checkbox"/> 1; Print "x ", rp 2, h N; Print "y ", rp 3, k N; </pre>	
	
<pre> x <input type="checkbox"/> 4.21213 y <input type="checkbox"/> 2.19203 </pre>	
<p><b>Configuración NO deformada para inicio de intercepción</b></p>	
<pre> ClearAll R0, RB, S0, S3, a11, a12, b1, b2; R <input type="checkbox"/> R0, 0, 0, R3; S <input type="checkbox"/> S0, 0, 0, S3; a11 <input type="checkbox"/> Ropq R, e1; a12 <input type="checkbox"/> Ropq S, e2; b1 <input type="checkbox"/> d1 a11; b2 <input type="checkbox"/> d2 a12; rp1 <input type="checkbox"/> Simplify b1, b2 N; </pre>	

<pre>f1 = rp1^2 - h; f2 = rp1^3 - k; f3 = R0^2 - R3^2 - 1; f4 = S0^2 - S3^2 - 1;</pre>	
<pre>sol = FindRoot[{f1, f2, f3, f4}, {R0, 0.1},               {R3, 0.6}, {S0, 0.1}, {S3, 0.6},               MaxIterations - 15]; resp = {R0, R3, S0, S3} /. sol; R0 = resp[[1]]; R3 = resp[[2]]; S0 = resp[[3]]; S3 = resp[[4]];</pre>	
<pre>g11 = Graphics[   AbsoluteThickness[6], RGBColor[0, 0, 1],   Line[h, k, b1, 2, h, b1 - 3, k],   AbsoluteThickness[3],   RGBColor[0.5, 0.5, 0.996109],   Line[b1 - 3, h, h1, 3, k],   rp1^2 - h, rp1^3 - k,   RGBColor[0.502, 0.251, 0.251],   Rectangle[h, 0.25, k,             h, 0.25, k, 0.3]; ];</pre>	
<pre>Show[g1, g11, Axes - True,       PlotRange - {0.1, 8, -2, 5},       AspectRatio - 1/1];</pre>	
<pre>Print["x = ", rp1^2, " h", N]; Print["y = ", rp1^3, " k", N]; Print["r1 = ", Chop[2 ArcSin R3, Degree]; Print["r2 = ",       Chop[2 ArcSin S3, Degree],       2 ArcSin R3, Degree];</pre>	
	

$x \square 2.$ $y \square 4.95159 \square 10^{13}$ $\square_1 \square 91.4688$ $\square_2 \square 10.2866$	
---	--

<b>Acoplamiento Robot particulas seguidoras movimiento relativo</b>	
<pre> ClearAll t, t2, t3, t1, g ; V0 □ 7; V1 □ 8; □0 □ 60 Degree; □1 □ 69 Degree; .original 102 g □ 9.81; t2 □ 0.7; t3 □ 1.3; num □ t3; t1 □ 0.3; step □ t3 26; .0.1s ax □ 2; .cambio de base para TPŞ bx □ 0; </pre>	
<pre> level1 □ Text FontForm "Intercepción Múltiple 2D", "Terminal", 16, 2.3, 4.9 ; leye2 □ Text FontForm "Interceptado ", "Terminal", 11, rr1 0.5, rr2 1 ; </pre>	
<pre> TPL1 □ Table PointSize 0.007, Point V0 Cos □0 t, V0 Sin □0 t - 0.5gt² t, 0, num, step ;  TPL2 □ Table PointSize 0.007, Point V1 Cos □1 t, V1 Sin □1 t - 0.5gt² t, 0, num, step ; </pre>	

<p><b>Algoritmo de intercepción</b></p> <pre> a  t1 t2 t1 t3 t2 t3 ; b  ax t t3 V1 Cos t t1    V1 Cos t1^2 c  t3^2 ax V0 Cos t0 t t1 t d  ax t V0 Cos t0 t t1 V0 Cos t0 t1^2 e  t^2 t1^2 t3 V0 Cos t0 V1 Cos t1 f  ax t t3 ax V1 Cos t1 t t1 ;  rr1 a t t3 b t2 c t d e t^2 f ; </pre>	
<pre> aa  t1 t2 t1 t3 t2 t3 ; bb  bxt b t t3 V1 Sin t t1    V1 Sin t1^2 cc  t3^2 bx V0 Sin t0 t t1 t dd  bxt V0 Sin t0 t t1 V0 Sin t0 t1^2 ee  t^2 t1^2 t3 V0 Sin t0 V1 Sin t1 ff  bxt t3 bx V1 Sin t1 t t1 ;  Y1 aa t t3 bb t2 cc t dd ee t^2 ff ; </pre>	
<pre>   2 t1 t2 t1 t3 ;   2 bxt 2bx gt t1 gt^2 t3 ;   2 bxt gt^2 t1 2bx gt^2 t3   gt^2 t t3 ;  rr2 Y1 t t2 ; </pre>	
<pre> caso1 Table PointSize 0.007, RGBColor 0,0,1 Point rr1,rr2 t,0,num step,step ; </pre>	

```

i = 0;
For t = 0, t = num_step, t = step,
g2 = Graphics TPL1, TPL2, casol,
If t = t2, level2, level1 =
Text FontForm "Tiempo absoluto: t" "t s",
"Courier", 12, 2.3, 4.5
Text FontForm
"Tiempo de Intercepción1: t2" "t2 s",
"Courier", 12, 2.3, 4.2
Text FontForm
"Tiempo de Intercepción2: t3" "t3 s",
"Courier", 12, 2.3, 3.9
Text FontForm "Tiempo de retardo: t1" "
t1 s", "Courier", 12, 2.3, 3.6

```

```

r1 = V0 Cos θ t;
r2 = V0 Sin θ t - 0.5gt^2;
r11 = V1 Cos θ1 t;
r22 = V1 Sin θ1 t - 0.5gt^2;

```

```

if t==t1,
a [ t1 t2 t1 t3 t2 t3 ;
b [ axt ax t3 V1 Cos t1 t3
V1 Cos t1 t1^2;
c [ t3^2 ax V0 Cos t1 t3;
d [ axt V0 Cos t1 t3 V0 Cos t1 t1^2;
e [ t^2 t1^2 t3 V0 Cos t1 V1 Cos t1 t3;
f [ axt t3 ax V1 Cos t1 t1;
rr1 [ a t t3 b t2 c t d e t^2 f ;

```

```

aa [ t1 t2 t1 t3 t2 t3 ;
bb [ bxt bx t3 V1 Sin t1 t3
V1 Sin t1 t1^2;
cc [ t3^2 bx V0 Sin t1 t3;
dd [ bxt V0 Sin t1 t3 V0 Sin t1 t1^2;
ee [ t^2 t1^2 t3 V0 Sin t1 V1 Sin t1 t3;
ff [ bxt t3 bx V1 Sin t1 t1;
Y1 [ aa t t3 bb t2 cc t dd ee t^2 ff ;

```

```

[ 2 t1 t2 t1 t3 ;
[ 2 bxt -2bx gt t1 gt t1^2 t3;
[ 2 bxt gt^2 t1 -2bx gt^2 t3
gt t1^2 t t3;
rr2 [ Y1 t t2 ;
, rr1 [ ax;
rr2 [ bx;

```

```

Text FontForm "Posición pinza robot x"
rr1 m, "Courier", 12, 2.6, 3.3
Text FontForm "Posición pinza robot y"
rr2 m, "Courier", 12, 2.6, 3.1
PointSize 0.02, RGBColor 1, 0, 0
Point r1, r2
PointSize 0.02, RGBColor 0, 0, 1
Point r1, r2
If t==t2, PointSize 0.02, RGBColor 1, 0, 0
Point r1, r2, Point rr1, rr2
;

```

CTINEMÁTICA INVERSA PARA EL ROBOT DE 2 GDL


<pre>ClearAll R, S, R0, R3, S0, S3, b1, b2, k1, k2, sol, f1, f2, f3, f4 ;</pre>	
<pre>R = R0, 0, 0, R3 ; S = S0, 0, 0, S3 ; k1 = Ropq R, e1 ; k2 = Ropq S, e12 ; b1 = d1 k1 ; b2 = d2 k2 ; rp2 = Simplify b1, b2 ; N;</pre>	
<pre>x = rr1 h ; y = rr2 k ; f1 = rp2 2 x ; f2 = rp2 3 y ; f3 = R0^2 R3^2 - 1 ; f4 = S0^2 S3^2 - 1 ;  iR0 = 0.7 ; iR3 = 0.1 ; iS0 = 0.7 ; iS3 = 0.1 ;  sol = FindRoot {f1, f2, f3, f4, R0, iR0, R3, iR3, S0, iS0, S3, iS3} ; MaxIterations = 15 ;</pre>	
<pre>e5 = 10^-5 ; Presición requerida If Abs x - rp2 2 sol &lt; e5 &amp;&amp; Abs y - rp2 3 sol &lt; e5, Null, Print "Error" ;  resp = R0, R3, S0, S3, sol ; R0 = resp 1 ; R3 = resp 2 ; S0 = resp 3 ; S3 = resp 4 ;</pre>	

<pre> gg Graphics AbsoluteThickness 6, RGBColor 0, 0, 1 Line h, k, b1, 2, h, k1, 3, k AbsoluteThickness 2, RGBColor 0, 0, 1 Line b1, 2, h, k1, 3, k, rp2, 2, h, rp2, 3, k RGBColor 0.502, 0.251, 0.251 Rectangle h, 0.25, k h, 0.25, k, 0.2 ; </pre> <p style="color: red;">.Fin de la CINEMÁTICA INVERSA.</p>	
<pre> Show gg, g2, Axes True, PlotRange {0.1, 5.5}, {0.5, 5} AspectRatio 1 1; </pre>	
<pre> i. ; U1 2 ArcSin R3 N; U2 2 ArcSin 53, 2 ArcSin R3 N;  ClearAll ;  0 0; f U1; 0d 0; fd 0; 0 0; f 0.1; a0 0; a1 0d; a2 <math>\frac{3}{f \cdot f'}</math> <math>\frac{2}{f \cdot 0d}</math> <math>\frac{1}{f \cdot fd}</math> N; a3 <math>\frac{2}{f \cdot f' \cdot f'}</math> <math>\frac{1}{f \cdot f}</math> <math>\frac{1}{fd \cdot 0d}</math> N; N;  p a0 a1 a2 a3 N; p1 Dt p, N; p2 Dt p1, N; </pre>	



<pre> Plot p, 1, 0, 0.1 PlotStyle RGBColor 0, 0, 1 Thickness 0.01 Frame True, FrameLabel "Tiempo, Posicion Rad Motor1, "", RotateLabel True, AxesStyle RGBColor 0, 0, 0 Thickness 0.01, GridLines Automatic </pre>	
<pre> Plot p1, 1, 0, 0.1 PlotStyle RGBColor 0, 0, 1 Thickness 0.01 Frame True, FrameLabel "Tiempo, Velocidad Rad s Motor1, "", RotateLabel True, AxesStyle RGBColor 0, 0, 0 Thickness 0.01, GridLines Automatic </pre>	
<pre> Plot p2, 1, 0, 0.1 PlotStyle RGBColor 0, 0, 1 Thickness 0.01 Frame True, FrameLabel "Tiempo, Aceleracion Rad s^2 Motor1, "", RotateLabel True, AxesStyle RGBColor 0, 0, 0 Thickness 0.01, GridLines Automatic </pre>	
<pre> i = 1 p = p1 p1 = p2 p2 = p1 If t &lt; t1, Print "Tiempo: ", t p = i p1 = 1 p2 = 2 Print "Tiempo Retardo" </pre>	
<pre> . Eslabón 1 Print "Eslabón 1" Print " ", 2*ArcSin R3, " Rad" Print " ", 2*ArcSin R3, " Degree, "°" Print " ", p Print " ", p1 Print " ", p2 </pre>	

<pre> Eslabón 2 Print "Eslabón 2"; Print "2", 2*ArcSin S3, 2*ArcSin R3, " Rad" ; Print "2", 2*ArcSin S3 Degree 2*ArcSin R3 Degree, "o" ; </pre>	
<pre> ClearAll []; [] 0 0; [] f U2; [] 0d 0; [] fd 0; [] 0 0; [] f 0.1; a0 [] 0; a1 [] 0d; a2 [] <math>\frac{3}{f \cdot f} \cdot \frac{2}{f \cdot 0} \cdot \frac{1}{f \cdot 0d} \cdot \frac{1}{fd \cdot N}</math>; a3 [] <math>\frac{2}{f \cdot f \cdot f} \cdot \frac{1}{f \cdot 0} \cdot \frac{1}{fd \cdot 0d \cdot N}</math>; pp [] a0 a1 a2 a3; pp1 [] Dt pp, N; pp2 [] Dt pp1, N; [] i pp; [] 1 i pp1; [] 2 i pp2; </pre>	
<pre> If t &lt; t1, Print "Tiempo:", t ; pp [] i 1 i ; pp1 [] 1 i 1 i ; pp2 [] 2 i 1 i ; Print "Tiempo Retardo" ; Print "2:", pp ; Print "2:", pp1 ; Print "2:", pp2 ; </pre>	
<pre> Plot pp, 1, 0, 0.1 PlotStyle [] RGBColor 0, 0, 1 Thickness 0.01 Frame [] True, FrameLabel [] "Tiempo, Posicion Rad, Motor2, "" RotateLabel [] True, AxesStyle [] RGBColor 0, 0, 0 Thickness 0.01 GridLines [] Automatic ; </pre>	

<pre>Plot pp1, [ , 0, 0.1 ] PlotStyle [ RGBColor 0, 0, 1 ] Thickness 0.01, Frame [ True, FrameLabel [ Tiempo, Velocidad Rad s, Motor2, "" ], RotateLabel True, AxesStyle [ RGBColor 0, 0, 0 ], Thickness 0.01 GridLines [ Automatic ]</pre>	
<pre>Plot pp2, [ , 0, 0.1 ] PlotStyle [ RGBColor 0, 0, 1 ] Thickness 0.01, Frame [ True, FrameLabel [ Tiempo, Aceleracion Rad s<sup>2</sup>, Motor2, "" ], RotateLabel True, AxesStyle [ RGBColor 0, 0, 0 ], Thickness 0.01 GridLines [ Automatic ]</pre> 	

---

## ANEXO B

A continuación se presenta el listado escrito en Matlab 7.0 release 14. que realiza la captura y el procesamiento de las imágenes para encontrar el centroide, el área y la orientación de objetos en movimiento.

```
%PROGRAMA PARA SEGUIMIENTO DE OBJETOS EN TIEMPO REAL.  
% TESIS DE DOCTORADO. MATLAB 7.0  
%Por: j. Alejandro Flores C.  
% 14 de marzo del 2005  
%
```

```
camara = videoinput('winvideo', 1, 'RGB24_160X120'); %320x240  
preview(camara)  
% Configura la cámara para que tome una imagen por intervalo de tiempo.  
set(camara,'FrameGrabInterval',1);  
% Accede al dispositivo durante de video.  
src=getselectedsource(camara);  
% El dispositivo se configura para que capture 30 frames por segundo.  
set(src,'FrameRate','30')  
% Prende la cámara para colocar las piezas y configurar el ambiente de  
% trabajo.  
start(camara)  
% Se genera un pause, hasta que se oprime una tecla  
pause
```

```
% Se captura la primera foto en formato RGB con 24 bits de profundidad  
BW = getsnapshot(camara);
```

```
% Se obtienen sus resoluciones en el eje X y en el eje Y.  
imageRes = camara.VideoResolution;  
imageWidth = imageRes(1);  
imageHeight = imageRes(2);
```

```
%Se muestra la imagen  
imshow(BW)  
pause
```

```
for m = 1:5
```

```
    BW = getsnapshot(camara);  
    % Se genera una imagen en escala de grises  
    I = rgb2gray(BW);  
    %Se binariza la imagen  
    BW = im2bw(I,0.25);  
    % limpia imagen de objetos con menos 100 pixeles  
    BW = bwareaopen(BW,100);
```

---

```

% Se obtiene la imagen complemento
BW = imcomplement(BW);
% Crea objeto estructurado morfológico en forma de disco
se = strel('disk',2);
% Genera un solo objeto
BW = imclose(BW,se);
% Se rellenan hoyos en los objetos para poder obtener el área, etc.
BW = imfill(BW,'holes');
imshow(BW)

pause

% Se dibujan los contornos de los objetos etiquetados
[B,L] = bwboundaries(BW,'noholes');
% Se despliega la matriz etiquetada y se dibuja cada contorno
imshow(label2rgb(L, @jet, [.5 .5 .5]))
hold on
for k = 1:length(B)
    boundary = B{k};
    plot(boundary(:,2), boundary(:,1), 'w', 'LineWidth', 2)
end

% ***** CALCULAR EL BORDE Y EL AREA Y EL CENTROIDE *****
stats = regionprops(L,'Area','Centroid','Orientation');
% Un for para cada uno de los bordes
for k = 1:length(B)
    % Se obtienen las coordenadas (X,Y) correspondientes a la etiqueta 'k'
    boundary = B{k};
    % Se obtiene el centroide
    centroid = stats(k).Centroid;
    % Se obtiene la orientación
    orientacion = stats(k).Orientation
    % Se obtiene el área
    area = stats(k).Area
    % Pone un tache en el centroide los objetos etiquetados
    plot(centroid(1),centroid(2),'yx','LineWidth',10);

    metric_string = sprintf('(%2.1f, %2.1f)\n %2.2f grados',centroid(),orientacion());

    x(:,k)=centroid(1);
    y(:,k)=centroid(2);

    text(boundary(1,2)-60,boundary(1,1)+35,metric_string,'Color','y',...
        'FontSize',14,'FontWeight','bold');
end
title(['Centroides y Orientaciones ',...
    'Regiones: Rectángulo y Circulo']);

% Grafica las posiciones de los centroides de la etiqueta k=1.

```

```
figure
plot(x(:,1),y(:,1),'--rs'),axis ij, axis equal, hold on;
xlabel('x');
ylabel('y');
title('Seguimiento de centroides');
hold on

end
```

PROGRAMA PARA CALCULAR EL PERFIL DE VELOCIDAD TRAPEZOIDAL CON  
MOVIMIENTO DE ENLACE PARABÓLICO \*.m (Matlab® 7.0)

```
q=[0 -7.41171];  
ac=[74858.3 74858.3];  
t0=0;  
tij=0.02; %tij=[2 1] ;  
[Qcoef,time,q_t,qd_t,qdd_t,intervs]=parab(q,ac,t0,tij,0.001);  
hold on  
subplot(3,1,1); plot(time,q_t,'k')  
subplot(3,1,2);plot(time,qd_t,'k')  
subplot(3,1,3);plot(time,qdd_t,'k')  
  
%savefile='prueba1.txt';  
%save(savefile,'time','qd_t')
```

---

```
%PROGRAMA PARA SEGUIMIENTO DE OBJETOS EN TIEMPO REAL no crítico.
% TESIS DE DOCTORADO. MATLAB 7.0
%Por: j. Alejandro Flores C. director: Dr. Luis Héctor Hernández
G.
```

```
% 14 de marzo del 2005
```

```
clear all,close all, clc;
clear all, clc
syms a t1 t2 t3 p1 p2 p3 p4 p5 p6 p7 p8 p9 p10 p11 p12;
syms plt1 plt2 plt3 plt4 plt5;

syms a t1 t2 dis_p;
syms q cont tiempo vel;

%%%%%%%%% ROBOT MITSUBISHI %%%%%%%%%%
a=serial('COM1','BaudRate', 9600,'Parity', 'even', 'StopBits' , 2,
'DataBits',7);
a.FlowControl='hardware';
fopen(a);
fprintf(a,'NT');
sprintf('HOME...Oprime tecla...')
pause()
fprintf(a,'GO');
fprintf(a,'TL 200'); % establecer el largo de la gripper TL 200 mm
pause(0.1);
fprintf(a,'WH');
pause(0.01);
p=fscanf(a)
disp('HOME DEL ROBOT');
pause();
disp('POSICIÓN PREPARATORIA...');

fprintf(a,'SP 7,L');
t0 = sprintf('MJ %0.1f,%0.1f,%0.1f,%0.1f,%0.1f',-60-90,0,0,0,-188);
fprintf(a,t0);
pause(0.5);
fprintf(a,'WH');
pause(0.01);
p=fscanf(a)
pause();
disp('POSICIÓN DE INICIO DE INTERCEPCIÓN...');
fprintf('MOVIMIENTO.. %2.0f',1);
%           Máx:   300, 130, 110, +-90,+-180
t1 = sprintf('MJ %0.1f,%0.1f,%0.1f,%0.1f,%0.1f',0,-50,-50,0,0);
fprintf(a,t1);
pause(0.01)
    pun = sprintf('HE 13');
    fprintf(a,pun)           %graba el punto
pause(0.1)
fprintf(a,'WH');
pause(0.01);
p=fscanf(a)
disp('listo...')
pause();

% SISTEMA DE VISION %%.
camara = videoinput('winvideo', 1, 'RGB24_160X120'); %320x240
```



```

    preview(camara)
% Configura la cámara que tome una imagen por intervalo de tiempo.
    set(camara,'FrameGrabInterval',1);
% Accede al dispositivo fuente de video.
    src=getselectedsource(camara);
% El dispositivo se configura para que capture 30 frames por segundo.
    set(src,'FrameRate','30')
% Prende la cámara para colocar las piezas y configurar el ambiente de
% trabajo.
    start(camara);
% Se genera una pausa, hasta que se oprime una tecla
% Se captura la primera foto en formato RGB con 24 bits de profundidad

% Se obtienen sus resoluciones en el eje X y en el eje Y.
    imageRes = camara.VideoResolution;
    imageWidth = imageRes(1)
    imageHeight = imageRes(2)
%Se muestra la imagen
    imshow(BW)

%pause
disp('opreme tecla para activar la cámara')
pause()
for m = 1:1

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% FUNCION CONVESPACIOS_ALE() %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

disp('Corriendo ciclo interno para detectar la pieza...')
%%Variables
q=0;
cont=0;
tiempo=0;
vel=0;
s=zeros(120,160);

while q<50000
I = getsnapshot(camara);
Inew=rgb2hsv(I);
% rect=[1 50 160 25];
% Inew = imcrop(Inew,rect);
%figure, imshow(Inew)
%pixval on
%h=zeros(120,160);
%i=zeros(120,160);
%%Componente s
s(:,:)=Inew(:,:,2);
%figure, imshow(s);
%pixval on;
% subplot(2,1,1),imshow(Inew),title('Imagen original')
% subplot(2,1,2),imshow(s),title('Componente s')

%Hacemos una segmentación por histograma
% figure,imhist(s)
for i=1:120
    for j=1:160
        if (s(i,j)<0.65)
            s(i,j)=0;
        end
    end
end

```

```

    end
end
%figure,imshow(s)
%pixval on
%% empieza ciclo interno cuando detecta la pieza.....

while max(s(:)>0.65 & s(67,155)==0)
tic()

%I=vfm('grab',1);
I = getsnapshot(camara);
Inew=rgb2hsv(I);
% rect=[1 50 160 25];
% Inew = imcrop(Inew,rect);
%Inew=rgb2hsv(Inew);
%h=zeros(120,160);
%i=zeros(120,160);
%%Componente s

s(:,:)=Inew(:,:,2);

%figure,imshow(s)
% figure,imhist(s)
% figure,
% subplot(2,1,1),imshow(Inew),title('Imagen original')
% subplot(2,1,2),imshow(s),title('Componente s')

%Hacemos una segmentaciòn por histograma
% figure,imhist(s)
for i=1:120
    for j=1:160
if (s(i,j)<0.65)
    s(i,j)=0;
end
    end
end
% imshow(s)
% pixval on

cont=cont+1;
tiempo=tiempo+toc();
sprintf('contador = %f tiempo= %f',cont,tiempo )
q=60000;
    end
%% ..... pieza que fue detectada .....
end

vel= (0.445)/tiempo;
disp('Fin...')

%%%%%%%%%% RESULTADOS %%%%%%%%%%%

sprintf('velocidad (m/s)= %f',vel)
disp('ya esta la foto!... escoge el pixel')
figure,imshow(I)
%rect=[1 50 160 25];
%BW = imcrop(I,rect);
%figure, imshow(BW)

```

```

pixval on

P = impixel(I,150,65);
%[27 80 111]
[BW,ind]=find_color(I,[],50,[P]);      %threshold original 90
imshow(BW);

% Se genera una imagen en escala de grises
BW = rgb2gray(BW);
imshow(BW);
    %[x,y,A,rect] = imcrop(I);
    %[x,y,II,xi,yi] = roipoly(BW)
    %[A,rect] = imcrop(BW)
    %II = roipoly(BW) %I
    %BW = edge(A, 'sobel', (graythresh(A) * .1));
    %BW = roicolor(A,0,80);
    %           imshow(BW)
    %Se binariza la imagen
    %imshow(A)
BW = im2bw(BW,0.1);
imshow(BW);

% limpia imagen de ojetos con menos 100 pixeles
BW = bwareaopen(BW,750);
    imshow(BW)
% Se obtiene la imagen complemento
BW = imcomplement(BW);
    imshow(BW)
    %pause()
%Crea objeto estructurado morfológico en forma de disco
se = strel('rectangle',[5 4]);
% Genera un solo objeto cerrado de imagen
BW = imclose(BW,se);
% Se rellenan hoyos en los objetos para poder obtener el área, etc.
BW = imfill(BW,'holes');
imshow(BW)

% Se dibujan los contornos de los objetos etiquetados
[B,L] = bwboundaries(BW,'noholes');
% Se despliega la matriz etiquetada y se dibuja cada contorno
imshow(label2rgb(L, @jet, [.5 .5 .5]))
hold on

for k = 1:length(B)
    boundary = B{k};
    plot(boundary(:,2), boundary(:,1), 'w', 'LineWidth', 2)
end
% ***** CALCULAR EL BORDE Y EL AREA Y EL CENTROIDE *****
stats = regionprops(L,'Area','Centroid','Orientation');
% Un for para cada uno de los bordes
for k = 1:length(B)
    % Se obtienen las coordenadas (X,Y) correspondientes a la etiqueta 'k'
    boundary = B{k};
    % Se obtiene el centroide
    centroid = stats(k).Centroid
    % Se obtiene la orientación
    orientacion = stats(k).Orientation
    % Se obtiene el área

```

```

area = stats(k).Area
% Pone un tache en el centroide los objetos etiquetados
plot(centroid(1),centroid(2), 'yx', 'LineWidth',10);
metric_string = sprintf('%2.1f, %2.1f)\n %2.2f
grados',centroid(),orientacion());
x(:,k)=centroid(1);
y(:,k)=centroid(2);
text(boundary(1,2)-40,boundary(1,1)+35,metric_string,'Color','y',...
      'FontSize',14,'FontWeight','bold');
end

title(['Centroides y Orientaciones Calculados']);

%Desplegar los valores de velocidad (v)y distancia (d)
dis_p = x(:,2)-x(:,1);
sprintf('velocidad = %f distancia (d): %f',vel,dis_p )
sprintf('RGB = %f',P)
%Grafica las posiciones de los centroides de la etiqueta k=1.
%figure
%plot(x(:,1),y(:,1),'--rs'),axis ij, axis equal, hold on;
xlabel('x');
ylabel('y');
%title('Seguimiento de centroides');
%hold on
        %fclose(a)
        %delete(a)
        %clear a
end
% para guardar la imagen:
%imwrite(fabric,'fabric.png','png');
%fabric = getsnapshot(vidobj);

%%%%%% ROBOT MITSUBISHI CONFIGURACION      %%%%%%

n=1;
%datos angulares valores originales:(10) t1=0.717 t2=-2.0687 t3= -1.66577
plt_1=[-32.4827 0.0 15.0508 25.0833 21.271 11.16 3.8365 -0.6351 -2.363 0.717
9.083];
plt_2=[-2.0001 0.0 6.3034 7+3.94473 -4.18373 -7.5284 -4.5979 0.3638 1.4956 -
4.0 -13.8447];
plt_3=[-35.8031 0.0 -12.1257 -0.4486 14.3036 17.8884 8.95196 -4.68797 -9.48675
-2.0 30.0327];

% INICIA SECUENCIA DE MOVIMIENTO 2
while n<12
    fprintf('MOVIMIENTO.. %2.0f',n);
    %pause(0.01)
    plt1 = plt_1(n);
    plt2 = plt_2(n);
    plt3 = plt_3(n);
    plt4 = -(plt_2(n)+plt_3(n));
    plt5 = plt_1(n);
    fprintf(a,'GO');
    pause(0.01);                                % PAUSA PARA AJUSTAR ENTRE PUNTOS

    if n==3
        fprintf(a,'SP 4,H');                    %fprintf(a,'MJ -30,0,0,0,0')
        fprintf(a,'GC');

```

```

    pause(0.01);
end

if n==4    %11
    fprintf(a, 'SP 9,H');          %fprintf(a, 'MJ -30,0,0,0,0')
    pause(0.01);
end

                %Máx:    300, 130, 110, +-90,+-180
t1 = sprintf('MJ %0.1f,%0.1f,%0.1f,%0.1f,%0.1f',plt1,plt2,plt3,plt4,plt5);
fprintf(a,t1);
pause(0.01);

pun = sprintf('HE %2.0f',n);
fprintf(a,pun)          %graba el punto
pause(0.01);
fprintf(a, 'WH');
pause(0.01);
p=fscanf(a)
% arreglo de valores leídos desde el bufer

%ps=strfind(p, ',')          % encuentra las comas ','
%plt1_l(n) = str2num(p(1:ps(1)-1))          % str2num convierte string a numero
%plt2_l(n) = str2num(p(ps(1)+1:ps(2)-1))
%plt3_l(n) = str2num(p(ps(2)+1:ps(3)-1))
%plt4_l(n) = str2num(p(ps(3)+1:ps(4)-1))
%plt5_l(n) = str2num(p(ps(4)+1:ps(4)+6))

n=n+1;
end

fprintf(a, 'SP 7,L');          %fprintf(a, 'MJ -30,0,0,0,0')
pause(0.01);
t1 = sprintf('MJ %0.1f,%0.1f,%0.1f,%0.1f,%0.1f',30,20,0,0,0);
fprintf(a,t1);
pause(0.01)

%          fprintf(a, 'NT');
%          disp('oprime tecla...1')
%          'pause();
%          fprintf(a, 'SP 4');
%          fprintf(a, 'MC 1,11');
%          disp('oprime tecla para leer posición...')
%          pause(0.01);
%          fprintf(a, 'WH');
%          pause(0.03);
%'          p=fscanf(a)
%          disp('cerrando el puerto de comunicaciones...')
%          fprintf(a, 'NT');
%          disp('FIN...')

fclose(a)
delete(a)
clear a

```

## FUNCION find\_color.m

```

function [image,index]=find_color(image,map,tr,colors)

%[IMAGE,IND]=FIND_COLOR(A,MAP,THRESHOLD,COLOR VECTOR)
%Encuentra el color de una imagen A y regresa la imagen,
%La imagen original con el píxel seleccionado se convierte e 0.
%IND, es el índice del píxel cambiado o modificado
%con el color seleccionado.
%
%MAP es la entrada de la imagen indexada, si la imagen no está indexada
%se dejan los brackets vacios ( [] ), TR es la tolerancia del color
%COLOR VECTOR contiene los valores del vector de color en RGB
%Ejemplo de una imagen:
%x=imread('football.jpg');
%subplot(2,1,1);imshow(x);
%title('Before');
%[image,ind]=find_color(x,[],90,[27 80 111]);
%subplot(2,1,2);imshow(image);
%title('After')
%figure
% y=imread('saturn.tif');
% image(328,438,:)=0;
% i= ~(im2bw(image,0.01));
% c=immultiply(y,i);
% c(:, :, 2)=c;
% c(:, :, 3)=c(:, :, 2);
% y(:, :, 2)=y;
% y(:, :, 3)=y(:, :, 1);
% C=imadd(image,c);
% imshow(C)

%esto es

I=make_rgb(image,map);

if isempty(colors)
disp(selecciona con el botón izquierdo del mouse el color deseado:') ;

colors=impixel(I);
close
end

[i j]=size(colors);
disp('espera....');
if i>1,
    colors=colors(1,:);
end
[image,index]= getcolor(I,colors,tr);
disp(';hecho!')

```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [image,index]=getcolor(I,color,tr)

[i j k]=size(I);
R=color(1,1)
G=color(1,2)
B=color(1,3)
I=double(I);
mask=( abs(I(:,:,1)-R) <tr ) & ( abs(I(:,:,2)-G) <tr ) & ( abs(I(:,:,3)-B) <tr
);
I(:,:,1)=I(:,:,1).*(~mask);
I(:,:,2)=I(:,:,2).*(~mask);
I(:,:,3)=I(:,:,3).*(~mask);

image=uint8(I);
index=find(mask==1);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function I=make_rgb(image,map)

[i j k]=size(image);

if (~isempty(map)),
    I=ind2rgb(image,map);
    I=im2uint8(I);

elseif (isempty(k)),
    I(:,:,2)=I;
    I(:,:,3)=I(:,:,1);

else
I=image;
end
```