

VICTOR HUGO MOSQUERA LEYTON



**CONTROL ROBUSTO DE UN ROBOT DE MÁS DE TRES
GRADOS DE LIBERTAD**

Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Maestría en Ingeniería
Área: Electrónica y Telecomunicaciones

Popayán
2008

VICTOR HUGO MOSQUERA LEYTON

**CONTROL ROBUSTO DE UN ROBOT DE MÁS DE TRES
GRADOS DE LIBERTAD**

Tesis presentada a la Facultad de Ingeniería
Electrónica y Telecomunicaciones de la
Universidad del Cauca para la obtención del
Título de Magíster en Ingeniería.

Área:
Electrónica y Telecomunicaciones.

Director:
Phd. Oscar Andrés Vivas Albán

Popayán
2008

A mi esposa e hijos

RESUMEN

Para el control de seguimiento de robots manipuladores existen esquemas, como el control PID y el control por par calculado que hacen que el error sea asintóticamente estable, El control PID es simple de implementar pero su desempeño disminuye al aumentar la velocidad de operación, por lo cual se desarrolló el control por par calculado; pero estas dos estrategias se ven muy afectadas en su desempeño cuando el modelo del manipulador no se conoce con exactitud, en este caso el control robusto es una buena opción ya que éste permite minimizar los efectos de las dinámicas no lineales desconocidas.

Spong propuso una ley de control no lineal robusta para robots manipuladores con incertidumbre paramétrica basada en la teoría de Lyapunov para asegurar la estabilidad de este. Esta técnica es muy utilizada en otras estrategias de control como son el control robusto adaptativo, control óptimo, etc. En esta tesis se retoma la ley de control propuesta por Spong y se aplica a un robot SCARA de 4 grados de libertad para el control de seguimiento de trayectorias.

En el área de la robótica el control *backstepping* es aplicado en robots móviles, robots manipuladores y robots paralelos, esta técnica de control garantiza la estabilidad asintótica y el seguimiento de trayectorias.

El aporte de esta tesis es la aplicación del control *backstepping* a robots manipuladores sin considerar la dinámica de los actuadores, algunos investigadores como Lofzazar, Egthesad y Chun-Yi emplean esta técnica considerando dicha dinámica, la cual les permite obtener un modelo representable según el enfoque *backstepping*. Teniendo en cuenta que no es muy sencillo identificar el modelo de los actuadores de un robot manipulador, este trabajo se enfocó en diseñar una ley de control *backstepping* partiendo únicamente de su modelo dinámico. La ley de control diseñada garantiza la estabilidad asintótica del manipulador frente a dinámicas no modeladas como es el modelo de los actuadores y a incertidumbre paramétrica.

La ley de control *backstepping* es aplicada a un robot SCARA de 4 grados de libertad y comparada con la ley de control propuesta por Spong y el control por par calculado mediante simulación.

Palabras clave: incertidumbre paramétrica, control robusto, backstepping, control de robots.

ABSTRACT

For the tracking control of robot manipulators approaches exist, such as PID control and computed torque control that make the mistake is asymptotically stable, PID control is simple to implement but their performance decreases with increasing speed of operation, so which is developed the control torque calculated, but the performance these two strategies decreases when the model of the manipulator is unknown exactly, in this case the robust control is a good choice because it allows minimizing the effects of nonlinear dynamics unknown.

Spong proposed a nonlinear robust control law to robot manipulators with parametric uncertainty based on the theory of Lyapunov to guaranteed stability. This scheme is used widely for other control strategies such as adaptive robust control, optimal control and so on. This thesis uses the control law proposed by Spong on a SCARA robot of 4 degrees of freedom to tracking control.

In robotics, backstepping control is used on mobile robots, manipulator robots and parallel robots, technique that guarantees asymptotic stability and tracking control.

The contribution of this thesis is the application of backstepping control in manipulator robots without including the dynamics of actuators. Many researchers as Lotfazar, Eghtesad and Chun-Yi, have been used this technique considering this dynamics, which allows express the model according to the backstepping approach. Taking into account that is not easy identify the actuator dynamics, the main work of this study is the design of a control law based only on its dynamic model. The designed control strategy guarantees asymptotic stability respect to non modeled dynamics (actuators model) and parametric uncertainty.

Backstepping control law is applied to a SCARA robot of 4 degrees of freedom and compared in simulation with the control law proposed by Spong and the computed torque control.

Key words: parametric uncertainty, robust control, backstepping, robot control.

TABLA DE CONTENIDO

	Pág.
CAPITULO 1	13
INTRODUCCIÓN	13
1.1 JUSTIFICACIÓN	13
1.2 CONTRIBUCIÓN DE LA TESIS	14
1.3 ORGANIZACIÓN DE LA TESIS	14
CAPITULO 2	17
MODELO GEOMÉTRICO, DINÁMICO Y CINEMÁTICO DEL ROBOT	17
2.1 MODELO GEOMÉTRICO	18
2.1.1 MODELO GEOMÉTRICO DIRECTO	20
2.1.2 MODELO GEOMÉTRICO INVERSO	20
2.2 MODELO CINEMÁTICO	22
2.2.1 MODELO CINEMÁTICO DIRECTO	22
2.2.2 MODELO CINEMÁTICO INVERSO	23
2.3 MODELO DINÁMICO	24
2.4 CARACTERÍSTICAS DEL ROBOT SCARA	28
CAPITULO 3	31
DISEÑO Y SIMULACIÓN DE CONTROL PID Y POR PAR CALCULADO DE UN ROBOT SCARA	31
3.1 CONTROL PID	32
3.1.1 SINTONIZACIÓN DEL CONTROLADOR PID	32
3.2 CONTROL POR PAR CALCULADO (CTC)	34
3.2.1 SINTONIZACIÓN DEL CONTROL CTC.	35
3.3 RESULTADOS DE SIMULACIÓN DEL CONTROL PID Y CTC	36
3.3.1 SIMULACIÓN DEL CONTROL PID	36
3.3.2 SIMULACIÓN DEL CONTROL CTC	40
3.3.3 ANÁLISIS DE RESULTADOS	41
CAPITULO 4	43
LEYES DE CONTROL BASADAS EN LYAPUNOV	43
4.1 TEORÍA DE LYAPUNOV	43
4.2 FUNCIONES DE CONTROL DE LYAPUNOV	46
4.3 BACKSTEPPING	48
4.3.1 BACKSTEPPING DE UN INTEGRADOR	48
4.3.2 BACKSTEPPING DE SISTEMAS EN REALIMENTACIÓN ESTRICTA	52
4.3.3 DISEÑO DE FCL POR AMORTIGUAMIENTO NO LINEAL	54
4.3.3.1 AMORTIGUAMIENTO NO LINEAL	54
4.3.4 BACKSTEPPING CON INCERTIDUMBRE	56

CAPÍTULO 5	57
DISEÑO DE LEYES DE CONTROL ROBUSTAS	57
5.1 LEY DE CONTROL ROBUSTA BASADA EN LYAPUNOV	58
5.1.1 LEY DE CONTROL ROBUSTA DE SPONG (SPONG, 1992).	59
5.2 LEY DE CONTROL BACKSTEPPING CON INCERTIDUMBRE	62
5.2.1 DISEÑO DE LEY DE CONTROL <i>BACKSTEPPING</i> CON INCERTIDUMBRE PARAMÉTRICA	63
CAPITULO 6	69
RESULTADOS DE SIMULACIÓN	69
6.1 ESQUEMA EN SIMULINK DE LAS ESTRATEGIAS DE CONTROL	69
6.1.1 ESQUEMA DE LA LEY DE CONTROL ROBUSTA DE SPONG	69
6.1.2 ESQUEMA DE LA LEY DE CONTROL <i>BACKSTEPPING</i> CON INCERTIDUMBRE.	71
6.2 RESULTADOS DE SIMULACIÓN.	72
6.2.1 RESULTADOS DE SIMULACIÓN CON INCERTIDUMBRE	72
6.2.2 RESULTADOS DE SIMULACIÓN PARA MAYOR VELOCIDAD DE EJECUCIÓN	78
6.2.3 ESFUERZO DE CONTROL	83
CAPITULO 7	91
CONCLUSIONES Y TRABAJOS FUTUROS	91
REFERENCIAS	93
ANEXO A	97
ANEXO B	121

LISTA DE TABLAS

	Pág.
Tabla 2.1: Parámetros geométricos	19
Tabla 2.2: Solución del MGI	21
Tabla 2.3: Parámetros de base simplificados del robot SCARA	27
Tabla 2.4: Valores máximos de posición articular	29
Tabla 2.5: Valores máximos de velocidad articular	29
Tabla 2.6: Parámetros inerciales de base del robot SCARA en estudio (Vivas, 2006)	30
Tabla 3.1: Constantes del controlador PID bloque <i>Signal Constrain</i> de <i>Simulink</i>	34
Tabla 3.2: Constantes del controlador CTC bloque <i>Signal Constrain</i> de <i>Simulink</i>	36
Tabla 3.3: Error de seguimiento del control PID y CTC	42
Tabla 6.1: Coeficientes de las matrices K y Λ	71
Tabla 6.2: Error de seguimiento, CTC, CRS y CBI, para un tiempo de ejecución de 3 segundos.	77
Tabla 6.3: Error de seguimiento, CRBL y <i>Backstepping</i> para tiempo de ejecución de 1 seg.	82
Tabla 6.4: Esfuerzo de control, CTC, CRS y CBI, para un tiempo de ejecución de 3 segundos.	86
Tabla 6.5: Esfuerzo de control, CTC, CRS y CBI para un tiempo de ejecución de 1 seg.	88

LISTA DE FIGURAS

	Pág.
Figura 2.1: Robot SCARA (Adep Technology, 2005)	18
Figura 2.2: Esquema del robot SCARA	19
Figura 2.3: SCARA Adept Cobra i800 (Adep Technology,2005)	29
Figura 3.1: Esquema control PID	33
Figura 3.2: Esquema del control CTC	35
Figura 3.3: Esquema del SCARA con control PID	38
Figura 3.4: Esquema del control PID	39
Figura 3.5: Trayectorias deseadas. a) Trayectoria circular. b) Trayectoria lineal	37
Figura 3.6: Error de seguimiento trayectoria circular control PID. a) Transitorio. b) Estacionario	39
Figura 3.7: Error de seguimiento para el cambio de dirección en trayectoria lineal control PID	40
Figura 3.8: Esquema del control CTC	40
Figura 3.9: Error de seguimiento trayectoria circular control CTC. a) Transitorio. b) Estacionario	41
Figura 3.10: Error de seguimiento para el cambio de dirección en trayectoria lineal control CTC	41
Figura 4.1: Diagrama del sistema (4.15) - (4.16)	49
Figura 4.2: Diagrama del sistema (4.21) - (4.22)	50
Figura 5.1: Esquema de la ley de control de Spong	63
Figura 5.2: Esquema control <i>backstepping</i>	67
Figura 6.1: Esquema en <i>Simulink</i> del control robusta de Spong	70
Figura 6.2: Esquema en <i>Simulink</i> del control <i>backstepping</i> con incertidumbre.	72
Figura 6.3: Error de seguimiento, trayectoria circular 5 seg. (transitorio) a) CRS, b) CBI	73
Figura 6.4: Error de seguimiento, trayectoria circular 5 seg. (estacionario) a) CRS, b) CBI	73
Figura 6.5: Error de seguimiento, trayectoria lineal 5 seg. (cambio de dirección). a) CRS b) CBI	74
Figura 6.6: Error de seguimiento, trayectoria circular 5 seg. (transitorio) con incertidumbre de 10% a) CRS b) CBI	74

Figura 6.7: Error de seguimiento, trayectoria circular 5 seg. (estacionario) con incertidumbre de 10% a) CRS b) CBI	75
Figura 6.8: Error de seguimiento, trayectoria lineal 5 seg. (cambio de dirección) con incertidumbre de 10% a) CRS, b) CBI	75
Figura 6.9: Error de seguimiento, trayectoria circular 5 seg. (transitorio) con incertidumbre de 20%. a) CRS b) CBI	76
Figura 6.10: Error de seguimiento, trayectoria circular 5 seg. (estacionario) con incertidumbre de 20%. a) CRS, b) CBI	76
Figura 6.11: Error de seguimiento, trayectoria lineal 5 seg. (cambio de dirección) con incertidumbre de 20%. a) CRS, b) CBI	77
Figura 6.12: Error de seguimiento, trayectoria circular 3 seg. (transitorio) con incertidumbre de 10%. a) CRS, b) CBI	79
Figura 6.13: Error de seguimiento, trayectoria circular 3 seg. (estacionario) con incertidumbre de 10%. a) CRS, b) CBI	79
Figura 6.14: Error de seguimiento, trayectoria lineal 3 seg. (cambio de dirección) con incertidumbre de 10%. a) CRS, b) CBI	80
Figura 6.15: Error de seguimiento, trayectoria circular 3 seg. (transitorio) con incertidumbre de 10%. a) CRS, b) CBI.	80
Figura 6.16: Error de seguimiento, trayectoria circular 3 seg. (estacionario) con incertidumbre de 10%. a) CRS, b) CBI.	81
Figura 6.17: Error de seguimiento, trayectoria lineal 3 seg. (cambio de dirección) con incertidumbre de 10%. a) CRS, b) CBI.	81
Figura 6.18: Par generado por los motores sin incertidumbre 5 seg. (transitorio) a) CRS, b) CBI.	83
Figura 6.19: Par generado por los motores sin incertidumbre 5 seg. (estacionario). a) CRS, b) CBI.	84
Figura 6.20: Par generado por los motores con incertidumbre del 10% 5 seg. (transitorio). a) CRS, b) CBI.	84
Figura 6.21: Par generado por los motores con incertidumbre del 10% 5 seg. (estacionario). a) CRS, b) CBI.	85
Figura 6.22: Par generado por los motores con incertidumbre del 20% 5 seg. (transitorio). a) CRS, b) CBI.	85
Figura 6.23: Par generado por los motores con incertidumbre del 20% 5 seg. (estacionario). a) CRS, b) CBI.	86
Figura 6.24: Par generado por los motores sin incertidumbre 5 seg. (cambio de dirección). a) CRS, b) CBI.	87
Figura 6.25: Par generado por los motores con incertidumbre del 10% 5 seg. (cambio de dirección). a) CRS, b) CBI.	87
Figura 6.26: Par generado por los motores con incertidumbre del 20% 5 seg. (cambio de dirección). a) CRS, b) CBI.	88

LISTA DE ABREVIATURAS

PID:	Control Proporcional Integral Derivativo
CTC:	Computed Torque Control
SCARA:	Selective Compliance Assembly Robot Arm
MGD:	Modelo Geométrico Directo
MGI:	Modelo Geométrico Inverso
Kg:	Kilogramos
ms	milisegundos
m:	Metros
MDD:	Modelo Dinámico Directo
GUAS:	Globally Uniformly Asymptotically Stable
GAS:	Globally Asymptotically Stable
Fcl:	Función de Control de Lyapunov
CRS:	Control Robusto de Spong
CBI:	Control <i>Backstepping</i> con Incertidumbre
N:	Newtons
CAD:	Computer Aided Design

LISTA DE SÍMBOLOS

- θ_j : ángulo entre los ejes x_{j-1} y x_j
 C_n : función coseno del ángulo n
 S_n : función seno del ángulo n
 ${}^{j-1}T_j$: matriz de transformación entre articulación $j-1$ y j
 q_n : posición de articulación n
 q_d : vector de posiciones deseadas
 x : vector
 x^T : transpuesta de vector x
 M : matriz
 M^{-1} : matriz Inversa
 \dot{M} : primera derivada de la matriz M
 \dot{q} : vector de velocidad
 \dot{q}_d : vector de velocidad deseado
 \ddot{q} : vector de aceleración
 \ddot{q}_d : vector de aceleración deseado
 \tilde{q} : vector de error de posición
 $\tilde{\dot{q}}$: vector de error de velocidad
 $\tilde{\ddot{q}}$: vector de error de aceleración
 \mathcal{R} : conjunto de los números reales
 $\|\bullet\|$: norma de \bullet
 $V(x)$: función de Lyapunov
 $\dot{V}(x)$: derivada de función de Lyapunov

CAPITULO 1

Introducción

1.1 Justificación

La disminución en el error de seguimiento de robots manipuladores en la realización de un trabajo es un objetivo importante en la robótica. En muchas aplicaciones industriales en las cuales intervienen los manipuladores, como por ejemplo las tareas de ensamble, se requiere que el efector final del robot siga una trayectoria con una alta precisión a alta velocidad (Renteria y Rivas, 2000). Actualmente las estrategias de control más comunes aplicadas a robots manipuladores de uso industrial son el control PID y el control por par calculado (CTC).

Generalmente los controladores tipo PID proveen soluciones aceptables en el entorno industrial, pero su desempeño se ve disminuido cuando las velocidades o aceleraciones aumentan, cuando se exige una mayor precisión, o se trabaja con cargas grandes, como respuesta a estos inconvenientes se ha diseñado el control por desacoplamiento no lineal, el cual suple en gran parte las deficiencias que un control PID posee. Pero este tipo de controlador, también llamado control por par calculado (CTC) y basado en la linealización por realimentación del sistema, tampoco es capaz de ofrecer las mejores respuestas en presencia de incertidumbre paramétrica o dinámicas no modeladas, ya que esta técnica parte de un modelado perfecto del manipulador para así lograr una linealización del modelo y posteriormente aplicar la ley de control.

Siendo los robots industriales sistemas no lineales y altamente acoplados, es necesario diseñar controladores avanzados que puedan aprovechar de mejor manera la dinámica del robot. Se hace imperativo entonces que los algoritmos de control sigan evolucionando, con el fin de ofrecer mejores desempeños a los cada vez mejores manipuladores que el mercado ofrece.

1.2 Contribución de la tesis

Las principales contribuciones de esta tesis son:

- Estudio y simulación de una ley de control robusta diseñada por Spong (Spong, 1992) y ampliamente utilizada para técnicas de control adaptativo, óptimo, etc. (Minf, 2004; Alonge, *et al*, 2004; Lin y Brandt, 1998).
- Aplicación del control *backstepping* robusta frente a incertidumbres paramétricas de un robot SCARA de 4 grados de libertad, sin tener en cuenta el modelo de los actuadores.
- Simulación de la ley de control *backstepping* propuesta, teniendo en cuenta la incertidumbre paramétrica y el aumento de la velocidad en el trabajo que debe realizar el manipulador.

El estudio y simulación de la ley de control diseñada por Spong fue publicada en la Conferencia Internacional de Control en Glasgow, Reino Unido en agosto de 2006.

- V. Mosquera, A. Vivas. “Robust control for a SCARA robot with parametric uncertainty”. Accepted to International Control Conference, ICC 2006, Glasgow, United Kingdom, august 2006.

1.3 Organización de la tesis

Esta tesis se encuentra organizada así:

Capítulo 2: En el cual se presenta el modelado geométrico, dinámico y cinemático del robot SCARA de 4 grados de libertad.

Capítulo 3: Este capítulo presenta el estudio y simulación de las estrategias de control PID y CTC en el seguimiento de dos trayectorias (circular y lineal) para el robot SCARA.

Capítulo 4: Los conceptos teóricos sobre la estabilidad de Lyapunov, funciones de control de Lyapunov y *backstepping* se presentan en este capítulo.

Capítulo 5: En este capítulo se presenta el estudio de la ley de control robusta diseñada por Spong y el diseño de la ley de control *backstepping* con incertidumbre paramétrica propuesta.

Capítulo 6: Los resultados de simulación de las técnicas de control CTC, control robusto de Spong y *backstepping*, para el seguimiento de las trayectorias circular y lineal y esfuerzos de control se consignan en este capítulo.

Capítulo 7: En este capítulo se presentan las conclusiones de los resultados presentados en el capítulo 6, en lo referente a error de seguimiento y esfuerzo de control.

Capítulo 2

Modelo geométrico, dinámico y cinemático del robot

Un robot manipulador es un sistema mecatrónico no lineal de varios grados de libertad y fuertemente acoplado (Dorf, 1988); además el robot es un planta que necesita ser controlada. En todas las aplicaciones de robots manipuladores se debe asegurar que el efector final ejecute un movimiento predeterminado. Para asegurar la correcta ejecución del movimiento, es necesario diseñar e implementar estrategias de control, las cuales entregarán las señales eléctricas a los actuadores de cada una de las articulaciones del manipulador con el fin de obtener el movimiento deseado (Spong, 1996).

Para lograr el diseño de estrategias de control es necesario conocer la configuración física del robot y la relación entre cada una de las estructuras que lo componen, lo cual se logra por medio del modelo geométrico; por otra parte también es necesario establecer la relación entre la velocidad del efector final y la velocidad de cada una de las articulaciones del manipulador mediante el modelo cinemático y por último, se deben conocer las ecuaciones de movimiento del robot manipulador las cuales relacionan las fuerzas aplicadas a cada una de las articulaciones con su posición, velocidad y aceleración las cuales se obtiene con el modelo dinámico (Khalil y Dombre, 2002; Sciavicco, Siciliano, 1996).

Este capítulo está orientado a obtener los modelos geométrico, cinemático y dinámico de un robot manipulador tipo *SCARA* de 4 grados de libertad como el que se muestra en la figura 1.1, con el fin de ser utilizado en la simulación de las estrategias de control que se detallarán en los capítulos siguientes.

El modelado de robot es un tema ampliamente desarrollado por varios autores y es un componente básico para el estudio de los manipuladores; los procedimientos presentados en este capítulo para obtener los modelos del robot *SCARA* se basan en los libros de Khalil, Sciavicco, Ollero y Barrientos (Khalil y Dombre, 2002; Sciavicco, Siciliano, 1996; Ollero, 2001; Barrientos, *et al.*, 1998).

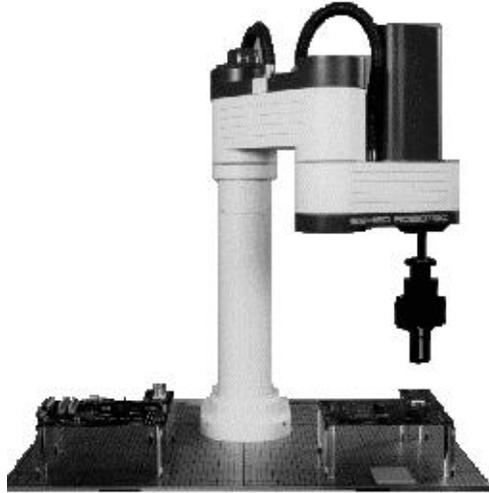


Figura 2.1: Robot SCARA (Adep Technology, 2005).

2.1 Modelo geométrico

Este modelo expresa la posición del efector final del robot en función de sus coordenadas articulares. Con el fin de desarrollar el modelo geométrico se aplicarán los métodos de Khalil-Kleininger y Craig (Craig, 1986). Para lo cual se debe tener en cuenta las siguientes consideraciones:

- El eje z_j es el eje de la articulación j
- El eje x_j es el eje común perpendicular al los ejes z_j y z_{j+1}

Donde j identifica la articulación del manipulador. Los parámetros geométricos involucrados en el modelo geométrico son:

- α_j : ángulo entre los ejes z_{j-1} y z_j correspondiente a una rotación alrededor de x_{j-1}
- d_j : distancia entre z_{j-1} y z_j a lo largo de x_{j-1}
- θ_j : ángulo entre los ejes x_{j-1} y x_j correspondiente a una rotación alrededor de z_j
- r_j : distancia entre x_{j-1} y x_j a lo largo de z_j

La variable articular q_j asociada a la j ésima articulación es θ_j o r_j según el tipo de articulación, rotoide o prismática respectivamente:

$$q_j = \bar{\sigma}_j \theta_j + \sigma_j r_j \quad (\sigma_j = 0 \text{ si la articulación es rotoide, } 1 \text{ si la articulación es prismática)}$$

La matriz de transformación permite expresar la posición del cuerpo R_j con respecto al R_{j-1} y se define por:

$${}^{j-1}T_j = \begin{bmatrix} C\theta_j & -S\theta_j & 0 & d_j \\ C\alpha_j S\theta_j & C\alpha_j C\theta_j & -S\alpha_j & -r_j S\alpha_j \\ S\alpha_j S\theta_j & S\alpha_j C\theta_j & C\alpha_j & r_j C\alpha_j \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.1)$$

La figura 2.2 muestra el esquema de robot SCARA, por medio del cual se identifica los parámetros geométricos y se consignan en la tabla 2.1.

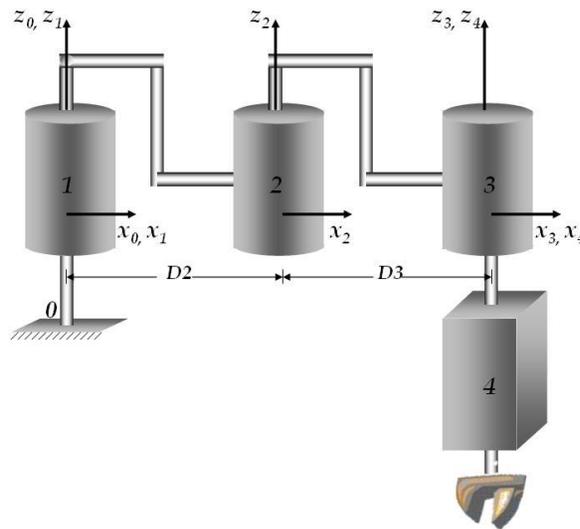


Figura 2.2: Esquema del robot SCARA.

Tabla 2.1: Parámetros geométricos

J	σ_j	α_j	d_j	θ_j	r_j
1	0	0	0	θ_1	0
2	0	0	D_2	θ_2	0
3	0	0	D_3	θ_3	0
4	1	0	0	0	r_4

Basado en la tabla de parámetros se calculan las matrices de transformación, teniendo en cuenta la ecuación (2.1). Las matrices son:

$$\begin{aligned}
{}^0T_1 &= \begin{bmatrix} C1 & -S1 & 0 & 0 \\ S1 & C1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & {}^1T_2 &= \begin{bmatrix} C2 & -S2 & 0 & D_2 \\ S2 & C2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
{}^2T_3 &= \begin{bmatrix} C3 & -S3 & 0 & D_3 \\ S3 & C3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & {}^3T_4 &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & r_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned} \tag{2.2}$$

Donde $Cn = \text{Cos}(\theta_n)$ y $Sn = \text{Sen}(\theta_n)$. Una vez calculadas las matrices de transformación se procede a obtener los modelos geométrico directo e inverso.

2.1.1 Modelo geométrico directo

El *modelo geométrico directo* (MGD) define la posición del efector final de robot en términos de las coordenadas articulares, esta relación se determina por medio de las matrices de transformación así:

$${}^0T_n = {}^0T_1(q_1) {}^1T_2(q_2) {}^2T_3(q_3) \dots {}^{n-1}T_n(q_n) \tag{2.3}$$

Teniendo en cuenta las matrices de transformación del robot SCARA (ecuación (2.2)) el modelo geométrico directo es:

$$U_0 = {}^0T_4 = \begin{bmatrix} C123 & -S123 & 0 & C12D3 + C1D2 \\ S123 & C123 & 0 & S12D3 + S1D2 \\ 0 & 0 & 1 & r_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.4}$$

El programa desarrollado en Matlab para el MGD (*mgd.m*) se muestra en el anexo A.

2.1.2 Modelo geométrico inverso

El *modelo geométrico inverso* (MGI) permite obtener todas las posibles soluciones del cálculo de las coordenadas articulares para una posición definida del efector final del robot. Dada la localización deseada del efector final U_0 definida en las ecuaciones (2.4) y (2.3), el MGI permite encontrar todas las posibles soluciones explícitas de (2.5) mediante el método de Paul (Paul, 1981).

$$U_0 = {}^0T_n = {}^0T_1(q_1) {}^1T_2(q_2) {}^2T_3(q_3) \dots {}^{n-1}T_n(q_n) \tag{2.5}$$

La solución de (2.5) se basa en multiplicar a ambos lados por la matriz ${}^jT_{j-1}$ así:

$$\begin{aligned}
 {}^1T_0U_0 &= {}^1T_2 {}^2T_3 {}^3T_4 \dots {}^{n-1}T_n = U_1 \\
 {}^2T_1U_1 &= {}^2T_3 {}^3T_4 \dots {}^{n-1}T_n = U_2 \\
 &\text{M} \\
 {}^{n-1}T_{n-2}U_{n-2} &= {}^{n-1}T_n
 \end{aligned} \tag{2.6}$$

Desarrollando el procedimiento explicado anteriormente se obtuvo el MGI del SCARA de 4 DOF, el cual se presenta en la tabla 2.2:

Tabla 2.2: Solución del MGI

Parámetros	Resultado
θ_1	$atan2(S1, C1)$
θ_2	$atan2\left(\pm\sqrt{1-(C2)^2}, C2\right)$
θ_3	$atan2(s_y, s_x) - \theta_2 - \theta_1$
r_4	P_z

Con

$$C2 = \frac{P_x^2 + P_y^2 - (D2)^2 - (D3)^2}{2D2D3}$$

$$S1 = \frac{B_1P_y - B_2P_x}{P_x^2 + P_y^2}$$

$$C1 = \frac{B_1P_x - B_2P_y}{P_x^2 + P_y^2}$$

$$B_1 = D2 + D3C2$$

$$B_2 = D3S2$$

En el anexo A se presenta el programa desarrollado en Matlab del MGI (*mgi.m*) utilizado en la simulación del robot SCARA.

2.2 Modelo cinemático

2.2.1 Modelo cinemático directo

El *modelo cinemático directo* relaciona la velocidad del efector final $\dot{\mathbf{x}}$ en función de la velocidad de las articulaciones del manipulador $\dot{\mathbf{q}}$; esta relación se expresa en la ecuación (2.7).

$$\dot{\mathbf{x}} = \mathbf{J}(\mathbf{q}) \dot{\mathbf{q}} \quad (2.7)$$

Donde $\mathbf{J}(\mathbf{q})$ es la matriz Jacobiana. Esta matriz provee la relación entre el desplazamiento diferencial del efector final del robot (dx) y la variación diferencial de la posición de las articulaciones (dq). Además la Jacobiana se aplica en robótica para el cálculo numérico de la solución del MGI y análisis del área de trabajo de los robots.

La matriz Jacobiana obtenida de un robot serial como lo es el SCARA es también denominada la matriz Jacobiana de base y se obtiene según la expresión (2.8).

$${}^i \mathbf{J}_{n:k} = \begin{bmatrix} \sigma_k {}^i a_k + \bar{\sigma}_k (-{}^k P_{ny} {}^i s_k + {}^k P_{nx} {}^i n_k) \\ \bar{\sigma}_k {}^i a_k \end{bmatrix} \quad (2.8)$$

Donde k representa la k ésima columna de ${}^n \mathbf{J}_n$. Los elementos de la columna ${}^n \mathbf{J}_{n:k}$ se calculan a partir de los elementos de la matriz ${}^k \mathbf{T}_i$, resultados intermedios obtenidos dentro del cálculo del MGD. Luego, el *modelo cinemático directo* se concentra en el cálculo de la Jacobiana, la cual se muestra en la ecuación (2.9):

$${}^0 \mathbf{J}_{4:k} = \begin{bmatrix} -D3S12 - D2S1 & -D3S12 & 0 & 0 \\ D3C12 + D2C1 & D3C12 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix} \quad (2.9)$$

De (2.7) y (2.9) se obtiene el modelo cinemático directo que se presenta en (2.10)

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix} = \begin{bmatrix} -D3S12 - D2S1 & -D3S12 & 0 & 0 \\ D3C12 + D2C1 & D3C12 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \\ \dot{\phi}_4 \end{bmatrix} \quad (2.10)$$

2.2.2 Modelo cinemático inverso

El *modelo cinemático inverso* entrega las velocidades $\dot{\phi}$ de las articulaciones, dadas las velocidades del efector final del robot \dot{x} , lo cual se representa en la ecuación (2.11).

$$\dot{\phi} = J^{-1} \dot{x} \quad (2.11)$$

Cuando la matriz J es de la forma:

$$J = \begin{bmatrix} A & 0 \\ B & C \end{bmatrix} \quad (2.12)$$

y si A y C son matrices cuadradas e invertibles, la inversa de la matriz Jacobiana es:

$$J^{-1} = \begin{bmatrix} A^{-1} & 0 \\ C^{-1}BA^{-1} & C^{-1} \end{bmatrix} \quad (2.13)$$

La matriz Jacobiana del robot *SCARA* (ecuación (2.9)) no es invertible debido a que el número de grados de libertad no coincide con la dimensión del espacio de trabajo (normalmente 6). Para el caso en que el número de grados de libertad es inferior a 6, la matriz Jacobiana tendrá más filas que columnas, esto indica que el movimiento del robot posee restricciones. Normalmente esto ocurre en robots diseñados para cierto tipo de trabajos, donde las restricciones no son importantes (tareas de ensamblaje donde son ideales los manipuladores tipo *SCARA*). En estos casos la orientación de la herramienta solo presenta movimiento en alguno de sus ejes; para el caso del robot *SCARA* la cuarta articulación presenta movimiento únicamente en el eje z, por lo cual se puede eliminar ese grado de libertad del espacio de trabajo, logrando así que la matriz Jacobiana sea cuadrada. Luego para simplificar la matriz Jacobiana se obvia las rotaciones del vector \dot{x} y tomamos solamente los valores de las articulaciones uno, dos y cuatro, obteniendo como resultado la matriz simplificada que se presenta en la ecuación (2.14).

$$\mathbf{J} = \begin{bmatrix} -D_3S_{12} - D_2S_1 & -D_3S_{12} & 0 \\ D_3C_{12} + D_2C_1 & D_3C_{12} & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.14)$$

Invirtiendo (2.14) se obtiene:

$$\mathbf{J}^{-1} = \begin{bmatrix} \frac{C_{12}}{D_2X} & \frac{S_{12}}{D_2X} & 0 \\ \frac{D_3C_{12} + D_2C_1}{D_3D_2X} & \frac{D_3S_{12} + D_2S_1}{D_3D_2X} & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad (2.15)$$

Con $X = -C_{12}S_1 + S_{12}C_1$

De (2.11) y (2.15) se obtiene el modelo cinemático inverso que se muestra en:

$$\begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \end{bmatrix} = \begin{bmatrix} \frac{C_{12}}{D_2X} & \frac{S_{12}}{D_2X} & 0 \\ \frac{D_3C_{12} + D_2C_1}{D_3D_2X} & \frac{D_3S_{12} + D_2S_1}{D_3D_2X} & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} \quad (2.16)$$

2.3 Modelo Dinámico

El *modelo dinámico inverso* es la relación entre las fuerzas aplicadas a los actuadores (Γ) y las posiciones, velocidades y aceleraciones articulares, y se expresa:

$$\Gamma = f(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, \mathbf{f}_e) \quad (2.17)$$

Donde f_e representa el esfuerzo exterior que ejerce el robot sobre el ambiente. Esta ecuación se llama modelo dinámico inverso. El *modelo dinámico directo* expresa las aceleraciones articulares en función de las posiciones, velocidades y fuerzas en las articulaciones.

$$\ddot{\mathbf{q}} = g(\mathbf{q}, \dot{\mathbf{q}}, \Gamma, \mathbf{f}_e) \quad (2.18)$$

Para la realización del modelado dinámico es necesario definir claramente los parámetros que se involucran en él; la notación a utilizar para este proceso es:

- G_j centro de gravedad del cuerpo C_j
 I_{G_j} tensor de inercia del cuerpo C_j con relación a una base paralela a R_j y de origen G_j
 I_{aj} momento de inercia del accionador j y de su reductor, sentido por la articulación
 jJ_j tensor de inercia del cuerpo C_j con relación a la base R_j , expresado así:

$${}^jJ_j = \begin{bmatrix} \int (y^2 + z^2) dm & -\int xy dm & -\int xz dm \\ -\int xy dm & \int (x^2 + z^2) dm & -\int yz dm \\ -\int xz dm & -\int yz dm & \int (x^2 + y^2) dm \end{bmatrix} = \begin{bmatrix} XX_j & XY_j & XZ_j \\ XY_j & YY_j & YZ_j \\ XZ_j & YZ_j & ZZ_j \end{bmatrix}$$

- M_j masa del cuerpo C_j
 MS_j primer momento de inercia del cuerpo C_j alrededor del origen de la base R_j , igual a $M_j S_j$. Es decir $[MX_j \quad MY_j \quad MZ_j]^T$ son los componentes de jMS_j
 MG_j momento de los esfuerzos exteriores ejercidos sobre el cuerpo C_j alrededor de G_j
 V_j velocidad del punto O_j
 \dot{V}_j aceleración del punto O_j
 ω_j velocidad de rotación del cuerpo C_j
 $\dot{\omega}_j$ aceleración de rotación del cuerpo C_j

Para obtener el modelo dinámico del robot se aplica el formalismo de Lagrange el cual describe las ecuaciones del movimiento en términos del trabajo y de la energía del sistema, lo cual se traduce cuando el esfuerzo sobre el órgano terminal es nulo, por la ecuación (2.19):

$$\Gamma_j = \frac{d}{dt} \frac{\partial L}{\partial \dot{q}_j} - \frac{\partial L}{\partial q_j} \quad (2.19)$$

Con:

L : lagrangiano del sistema igual a: $L = E - U$.

E : energía cinética total del sistema.

U : energía potencial total del sistema.

La expresión de la energía cinética involucra las velocidades articulares del manipulador, así:

$$E = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{M} \dot{\mathbf{q}} \quad (2.20)$$

Donde \mathbf{M} es la **matriz de inercia** del robot o matriz de energía cinética. Sus elementos son función de las variables articulares \mathbf{q} .

La fuerza total puede entonces escribirse como:

$$\Gamma = \mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} + \mathbf{Q}(\mathbf{q}) \quad (2.21)$$

Con:

$\mathbf{Q}(\mathbf{q})$: vector de fuerzas de gravedad.

$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}}$: Vector que representa las fuerzas de Coriolis y centrífugas tal que

$$\mathbf{C} \dot{\mathbf{q}} = \mathbf{M} \ddot{\mathbf{q}} - \frac{\partial E}{\partial \mathbf{q}} \quad (2.22)$$

Varias formas pueden utilizarse para la matriz \mathbf{C} , por ejemplo utilizando el símbolo de Christophell $C_{i,jk}$:

$$\left\{ \begin{array}{l} C_{ij} = \sum_{k=1}^n C_{i,jk} \dot{q}_k \\ C_{i,jk} = \frac{1}{2} \left[\frac{\partial A_{ij}}{\partial q_k} + \frac{\partial A_{jk}}{\partial q_j} - \frac{\partial A_{jk}}{\partial q_i} \right] \end{array} \right. \quad (2.23)$$

Y los elementos del vector \mathbf{Q} son calculados por la expresión (2.24).

$$Q_i = \frac{\partial U}{\partial q_j} \quad (2.24)$$

Los elementos \mathbf{M} , \mathbf{C} y \mathbf{Q} son funciones de los parámetros geométricos e inerciales del mecanismo, para el cálculo de estas matrices es necesario inicialmente calcular las energías cinética y potencial de todos los cuerpos del robot. Se procede así:

- El elemento M_{ij} es igual al coeficiente de $(\dot{q}_i^2/2)$ de la expresión de la energía cinética, y el elemento M_{ij} , si $i \neq j$, es igual al coeficiente de $\dot{q}_i \dot{q}_j$.
- El cálculo de \mathbf{C} se hace por medio del símbolo de Christophell (Ecuación (2.23)).
- El cálculo de \mathbf{Q} se hace con la expresión (2.24).

Los parámetros inerciales de base o parámetros identificables constituyen el conjunto mínimo de parámetros con los cuales se puede calcular el modelo dinámico de un robot, la determinación de los parámetros inerciales de base son esenciales para la identificación de los parámetros dinámicos, en otras palabras, para obtener el modelo dinámico de un robot manipulador es necesario identificar los parámetros inerciales del mismo; dichos parámetros pueden ser deducidos a partir de los parámetros inerciales estándar:

- M_j : masa de la articulación j
- jMS_j : vector columna del primer momento de la articulación j con respecto a la referencia R_j
- jJ_j : matriz del tensor de inercia de la articulación j con respecto a la referencia R_j
- Ia_j : matriz diagonal de las inercias de los actuadores de la articulación j

Algunos parámetros que no tienen efectos sobre el modelos dinámico se eliminan y otros se agrupan a partir del algoritmo presentado en Khalil (Khalil y Dombre, 2002) lo que permite trabajar sólo con 11 parámetros inerciales de base, ya que los 33 restantes se hacen cero debido a la reagrupación, tal como se puede observar en la Tabla 2.3

Tabla 2.3: Parámetros de base simplificados del robot SCARA

j	XX _j	XY _j	XZ _j	YY _j	YZ _j	ZZ _j	MX _j	MY _j	MZ _j	M _j	IA _j
1	0	0	0	0	0	ZZR1	0	0	0	0	0
2	0	0	0	0	0	ZZR2	MXR2	MY2	0	0	IA2
3	0	0	0	0	0	ZZR3	MXR3	MYR3	0	0	IA3
4	0	0	0	0	0	0	0	0	0	0	IA4

El *modelo dinámico inverso* se presenta de forma completa en la ecuación (2.25).

$$\Gamma = \mathbf{M}(q)\ddot{q} + \mathbf{C}(q, \dot{q})\dot{q} + \mathbf{Q}(q) \quad (2.25)$$

Donde \mathbf{M} es la matriz de inercia, \mathbf{C} es la matriz de fuerzas de coriolis y centrípetas y \mathbf{Q} es el vector de fuerzas de gravedad. La matriz de inercia se calcula a partir de la energía cinética y potencial de cada articulación del robot, la cual contiene los términos reagrupados gracias a los parámetros de base. Las fuerzas de coriolis y centrípeta no se tendrán en cuenta en el modelo ya que para este caso las velocidades son bajas, por lo cual estas fuerzas no presentan alta incidencia en el comportamiento del robot, y calculando la matriz de inercia y el vector de fuerzas de gravedad el modelo dinámico del robot es:

$$\Gamma = M\ddot{q} + Q \quad (2.26)$$

$$= \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \ddot{q} + Q$$

Donde

$$\begin{aligned} a_{11} &= ZZR1 + ZZR2 + 2MXR2C2D2 - 2MY2D2S2 + \\ & 2MXR3(-S2S3D2 + C3C2D2 + D3C3) + \\ & 2MYR3(-S2C3D2 - D2C2S3 - D3S3) + \\ & ZZ4 + ZZR3 \\ a_{12} &= 2ZZR2 + 2MXR2C2D2 - 2MY2S2D2 + 2ZZR3 + \\ & 2MXR3[(-S2S3D2 + C3C2D2 + D3C3) + D3C3] - \\ & 2MYR3[(S2C3D2 + D2C2S3 + D3S3) + D3S3] + ZZ4 \\ a_{13} &= ZZR3 + MXR3(-S2S3D2 + C3C2D2 + D3C3) - \\ & MYR3(S2C3D2 + D2C2S3 + D3S3) + ZZ4 \\ a_{22} &= IA2 + ZZR2 + ZZR3 + 2MXR3D3C3 - \\ & 2MYR3D3C3 + ZZ4 \\ a_{23} &= ZZR3 + MXR3(D3C3) - 2MYR3(D3S3) + ZZ4 \\ a_{33} &= IA3 + ZZR3 + ZZ4 \\ a_{44} &= IA4 \\ a_{14} &= a_{24} = a_{43} = 0 \\ Q_1 &= Q_2 = Q_3 = 0 \\ Q_4 &= -G3M4 \end{aligned}$$

El *Modelo dinámico inverso* del robot SCARA se programó en Matlab, este programa se encuentra en el anexo A.

2.4 Características del robot SCARA

Los parámetros base utilizados para el robot se extrajeron de Vivas 2006 (Vivas, 2006), complementadas sus características con ayuda del manual del robot Adept Cobra i800 (Adept Technology, 2005). La figura 1.3 presenta el esquema del robot Adept Cobra 800i y sus dimensiones físicas expresadas en metros. En la disposición de los eslabones del robot Adept Cobra 800i se encuentra que la articulación tres corresponde a la articulación prismática.

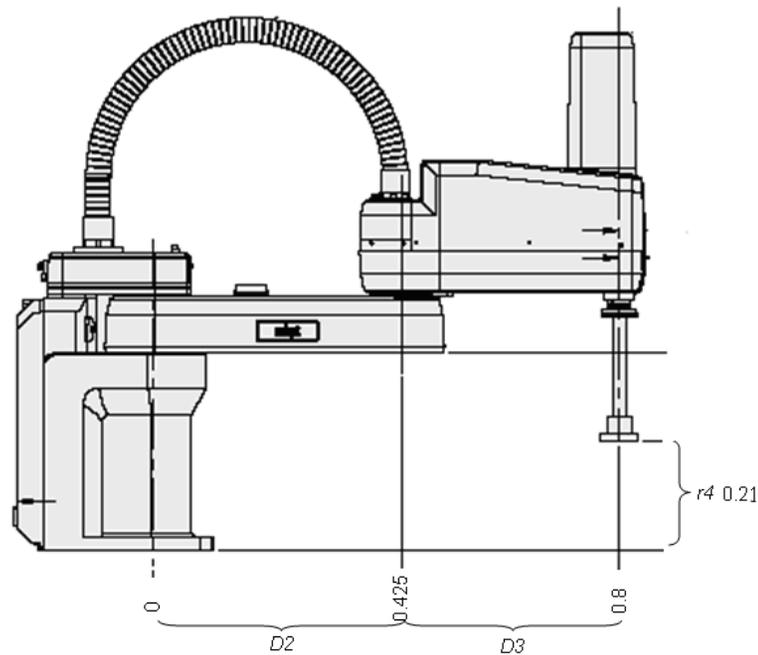


Figura 2.3: SCARA Adept Cobra i800 (Adep Technology,2005).

Las tablas 2.4 y 2.5 presentan los valores máximos de posición y velocidad respectivamente; en la tabla 2.6 se presentan los valores de los parámetros de base del robot SCARA.

Tabla 2.4: Valores máximos de posición articular

Articulación	Posición articular máxima
1	$\pm 105^{\circ}$
2	$\pm 175.5^{\circ}$
3	0.21m
4	$\pm 360^{\circ}$

Tabla 2.5: Valores máximos de velocidad articular

Articulación	Velocidad articular máxima
1	386 ^o /seg
2	720 ^o /seg
3	1.1 m/seg
4	1200 ^o /seg

Tabla 2.6: Parámetros inerciales de base del robot SCARA en estudio (Vivas, 2006)

Parámetro	Valor
<i>ZZR1</i>	3.38
<i>ZZR2</i>	0.066
<i>ZZR3</i>	0.1
<i>MXR2</i>	0.242
<i>MXR3</i>	0.2
<i>MY2</i>	0.001
<i>MYR3</i>	0.1
<i>IA2</i>	0.045
<i>IA3</i>	0.045
<i>IA4</i>	0.045
<i>M4</i>	0.5

Las unidades para los elementos del tensor de inercia son $Kg.m^2$, para el primer momento de inercia son $Kg.m$, para la inercia del actuador $Kg.m^2$ y para las masas Kg .

Capitulo 3

Diseño y simulación de control PID y por par calculado de un robot SCARA

El control de robots ha sido estudiado por mucho tiempo y las técnicas comúnmente utilizadas son el control Proporcional, Integral, Derivativo (PID), el cual considera al sistema como un sistema lineal y cada articulación es controlada por un control descentralizado; y el Control por Par Calculado o *CTC* (*Computed Torque Control*) que linealiza y desacopla el modelo del robot. El control PID tiene la ventaja de su fácil implementación, pero su desventaja es que al incrementar la velocidad de operación del robot, este pierde precisión en el seguimiento de trayectorias y debido a que las ecuaciones que representan la dinámica de los robots son fuertemente acopladas y altamente inciertas su sintonización es mas compleja que la sintonización del control *CTC* (Li; *et al.*, 2001). Por otra parte el *CTC* es utilizado cuando se requiere que el robot cumpla con tareas a altas velocidades y precisión; esta técnica de control requiere el cálculo en línea del modelo dinámico inverso del robot, además el control *CTC* convierte el problema de control no lineal acoplado en un problema de control lineal (Khalil y Dombre, 2002; Sciavicco y Siciliano, 1996; Barrientos, *et al.*, 1998).

En este capítulo se desarrollarán los controles PID y *CTC*, su respectiva sintonización para el robot SCARA de 4 grados de libertad modelado en el anterior capítulo y se mostrarán los resultados de simulación obtenidos de la aplicación de estas dos técnicas de control.

Como se mencionó anteriormente el control PID y *CTC* son técnicas ampliamente utilizadas en el campo de los robots manipuladores y se han convertido en un conocimiento fundamental en el estudio de la robótica y son claramente desarrollados por Khalil, Sciavicco y Barrientos (Khalil y Dombre, 2002; Barrientos, *et al.*, 1998; Sciavicco y Siciliano, 1996) y a su vez fueron tomados como referencias para el desarrollo de este capítulo.

3.1 Control PID

En la actualidad la mayoría de los robots industriales hacen uso del control PID descentralizado con ganancias constantes. El control es implementado en cada una de las articulaciones de manera simple y presenta un bajo costo computacional. Sin embargo, para aplicaciones de alta velocidad no se obtiene buen desempeño dinámico, por ello se buscan nuevas alternativas que lleguen a suplir esta desventaja si la aplicación lo requiere.

Por otra parte al aplicar un controlador PID es difícil alcanzar algunos niveles deseados en el desempeño del controlador ya que las ecuaciones que representan la dinámica de los robots son fuertemente acopladas y pueden llegar a ser altamente inciertas (Li, *et al.*, 2001).

Como su nombre lo indica el algoritmo de control PID consta de tres modos básicos: el término proporcional P que efectúa una acción de control correctiva proporcional al error, el término integral I entrega una corrección proporcional a la integral del error asegurando que el esfuerzo de control sea suficiente para reducir el error de estado estacionario a cero. Sin embargo, la acción integral tiende a tener un efecto negativo sobre la estabilidad y el término derivativo D , aplica una acción de control proporcional a la razón de cambio del error, presentando un efecto estabilizador en el control (Goodwin, *et al.*, 2000).

El control PID está dado por la expresión (3.1) y el esquema presentado en la figura 3.1 muestra el diagrama en bloques de esta técnica.

$$\Gamma = K_p(\mathbf{q}_d - \mathbf{q}) + K_d(\dot{\mathbf{q}}_d - \dot{\mathbf{q}}) + K_i \int_{t_0}^t (\mathbf{q}_d - \mathbf{q}) d\tau \quad (3.1)$$

Donde \mathbf{q}_d : Vector de posiciones deseadas.
 $\dot{\mathbf{q}}_d$: Vector de velocidades deseadas.
 K_p , K_d y K_i : matrices ($n \times n$) diagonales definidas positivas (n representa el número de grados de libertad del manipulador).

3.1.1 Sintonización del controlador PID

La sintonización del controlador PID se basa en definir las matrices K_p , K_i y K_d , las cuales son matrices $n \times n$ definidas positivas y diagonales, donde los elementos de la diagonal corresponden a las ganancias de controlador para cada articulación. Se puede considerar que cada una de las articulaciones del robot es modelada por medio de una ecuación diferencial de segundo orden, como lo indica la ecuación (3.2).

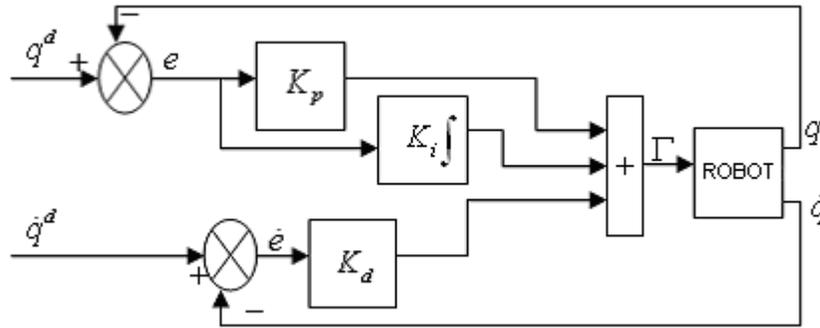


Figura 3.1: Esquema control PID.

$$\Gamma_j = a_j \ddot{q}_j + F_{vj} \dot{q}_j + \gamma_j \quad (3.2)$$

Donde a_j es el máximo de los valores de la diagonal principal de la matriz \mathbf{M} , F_{vj} es la fuerza de frotamiento viscoso, la cual se presenta en presencia de movimiento y γ_j representa el par de disturbio. Asumiendo que γ_j es igual a cero, entonces la función de transferencia en lazo cerrado está dada por:

$$\frac{q(s)}{q_j^d(s)} = \frac{K_{dj}s^2 + K_{pj}s + K_{ij}}{a_j s^3 + (K_{dj} + F_{vj})s^2 + K_{pj}s + K_{ij}} \quad (3.3)$$

Para obtener el valor de las ganancias del controlador PID se busca que la ecuación característica $Q(s)$ posea un polo triple con parte real negativa asegurando así la estabilidad del sistema en lazo cerrado. Luego la ecuación característica deseada es:

$$Q(s) = a_j (s + w_j)^3 \quad (3.4)$$

Donde w_j es la mitad de la frecuencia de resonancia (w_{rj}) de la articulación j . Con el fin de obtener las constantes del controlador PID, se hace la función característica del sistema en lazo cerrado igual a la ecuación (3.4), de lo cual se obtiene:

$$K_{pj} = 3a_j w_j^2 \quad (3.5)$$

$$K_{dj} + F_{vj} = 3a_j w_j \quad (3.6)$$

$$K_{ij} = a_j w_j^3 \quad (3.7)$$

El método de sintonización por medio de la frecuencia de resonancia del control PID que se expone en Khalil y Dombre (Khalil y Dombre, 2002) se aplicó al robot SCARA (los resultados se muestran en el anexo B). Debido a que no fueron completamente satisfactorios se realizó una sintonización con ayuda del bloque de *Simulink Signal*

Constrain, el cual realiza un proceso iterativo que consisten en ejecutar el modelo en *Simulink* basado en unos parámetros de optimización y en requerimientos de desempeño. En cada iteración se sintonizan las constantes del controlador y se almacenan los valores seleccionados y la respuesta del modelo (error de seguimiento), comparándose la respuesta obtenida con la deseada y con las obtenidas en las iteraciones pasadas; este proceso continua hasta que se encuentran los parámetros que hacen que el sistema tenga la respuesta deseada. El proceso para la aplicación del bloque *Signal Constrain* se puede observar en (The MathWorks, 2008).

Una vez realizado el proceso de sintonización del control PID con ayuda de *Simulink* (bloque *Signal Constrain*), los resultados obtenido por este procedimiento se muestran en la tabla 3.1. En el anexo B se puede observar el proceso de sintonización con el bloque *Signal Constrain*.

Tabla 3.1: Constantes del controlador PID bloque *Signal Constrain* de *Simulink*

Articulación	K_p	K_i	K_d
1	220000	1000	150
2	280000	1000	170
3	80000	1000	120
4	200000	1000	250

3.2 Control por par calculado (CTC)

El control por desacoplamiento lineal o Control por Par Calculado (CTC) mejora el desempeño del robot en lo referente a la disminución del tiempo de ejecución de tareas sin perder precisión (Khalil y Dombre, 2002; Sciavicco y Siciliano, 1996; Barrientos, *et al.*, 1998); esta estrategia de control posee un mayor costo computacional a diferencia del PID debido a que es necesario calcular el modelo dinámico del manipulador en línea. Sin embargo con los computadores utilizados hoy en día esto no ofrece ninguna dificultad para la implementación de este tipo de controlador.

La expresión (3.8) presenta la forma general de la ecuación dinámica de un robot manipulador

$$\Gamma = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + Q(q) \quad (3.8)$$

La ecuación (3.8) se puede escribir de forma compacta así;

$$\Gamma = M(q)\ddot{q} + H(q, \dot{q}) \quad (3.9)$$

Se asume que las posiciones y velocidades de las articulaciones son medibles y sin ruido, además sean \hat{M} y \hat{H} las estimaciones respectivas de M y H . Partiendo de la forma compacta de la ecuación del modelo dinámico, se define una ley de control de la forma:

$$\Gamma = \hat{M}(q)w(t) + \hat{H}(q, \dot{q}) \quad (3.10)$$

Sustituyendo (3.10) en (3.9) y en condiciones ideales, en un perfecto modelado y en ausencia de disturbios, se tiene que \hat{M} igual a M y \hat{H} igual a H . El problema se reduce entonces a un control lineal de n ecuaciones lineales desacopladas de segundo orden (doble integrador).

$$\ddot{q} = w(t) \quad (3.11)$$

Donde $w(t)$ es el nuevo vector de control y se define mediante la siguiente ecuación:

$$w(t) = \ddot{q}_d + K_v(\dot{q}_d - \dot{q}) + K_p(q_d - q) \quad (3.12)$$

Donde K_p y K_v son las matrices ($n \times n$) diagonales y definidas positivas. En la figura 3.2 se presenta el esquema del control por par calculado.

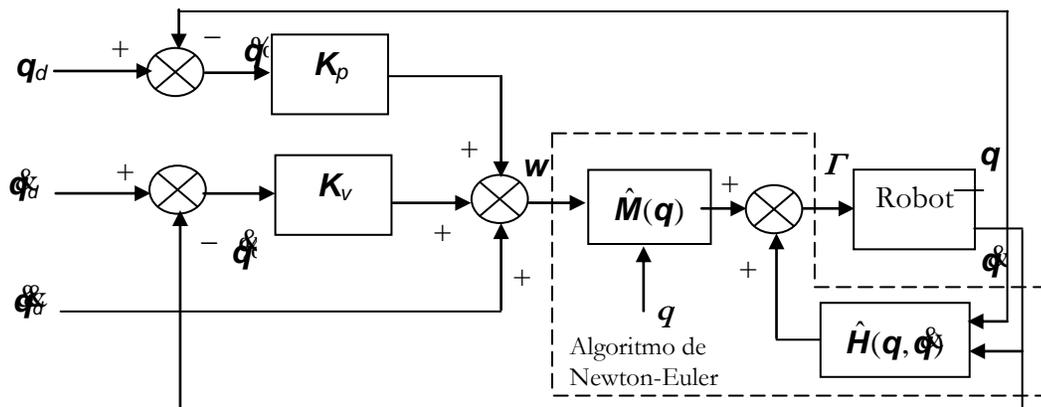


Figura 3.2: Esquema del control CTC.

3.2.1 Sintonización del control CTC.

La sintonización del control CTC se realizó por el método de la frecuencia de resonancia (Anexo B); al igual que para el control PID, los resultados no fueron satisfactorios, realizándose igualmente la sintonización con ayuda del bloque de *Signal Constrain* de *Simulink*. La tabla 3.1 muestra los resultados obtenidos en este procedimiento,

Tabla 3.2: Constantes del controlador CTC bloque *Signal Constrain* de *Simulink*

Articulación	K_p	K_v
1	320000	1200
2	280000	850
3	200000	1200
4	120000	700

3.3 Resultados de simulación del control PID y CTC

3.3.1 Simulación del control PID

La figura 3.3 muestra el esquema general del robot SCARA controlado por el PID desarrollado en *Simulink*; este esquema se encuentra compuesto por:

- Las trayectorias (cons1, cons2 y cons3). Ellas son generadas por los programas en Matlab *circular.m* o *lineal.m*
- El Modelo Geométrico Inverso (MGI), el cual realiza la conversión de coordenadas cartesianas a articulares, *mgj.m* es el programa que ejecuta este modelo
- El controlador PID, cuyo esquema se presenta en la figura 3.4 y las constantes del mismo se generan por el programa de Matlab *Inicio.m*
- El Modelo Dinámico Directo (MDD), por medio del cual se simula el comportamiento del SCARA debido a las trayectorias deseadas, el programa que simula este modelo es *scara_directo.m*
- El Modelo Geométrico Directo (MGD), el cual realiza la transformación de coordenadas articulares a cartesianas, el programa se denomina *mgd.m*
- La función *diferencia.m* que realiza el cálculo del error cartesiano entre las trayectorias deseadas y las realizadas por el robot.

Todos los programas mencionados se encuentran consignados en el anexo A.

El programa *inicio.m* carga la trayectoria deseada y los valores de las constantes del controlador, los cuales se presentaron en la tabla 3.4, Las trayectorias que se utilizaron para la simulación se muestran en la figura 3.5, las cuales se ejecutan en 5 seg.

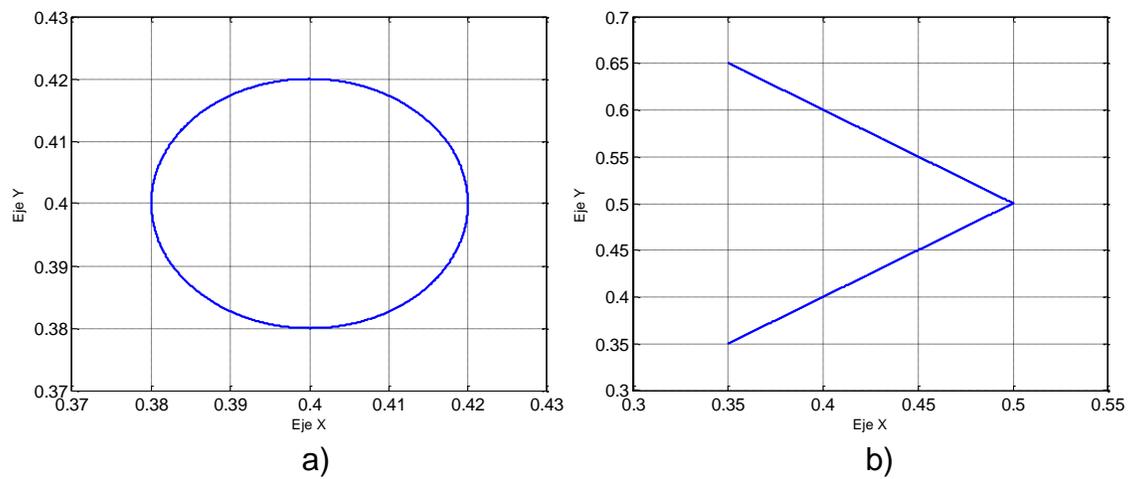


Figura 3.5: Trayectorias deseadas.
a) Trayectoria circular. b) Trayectoria lineal

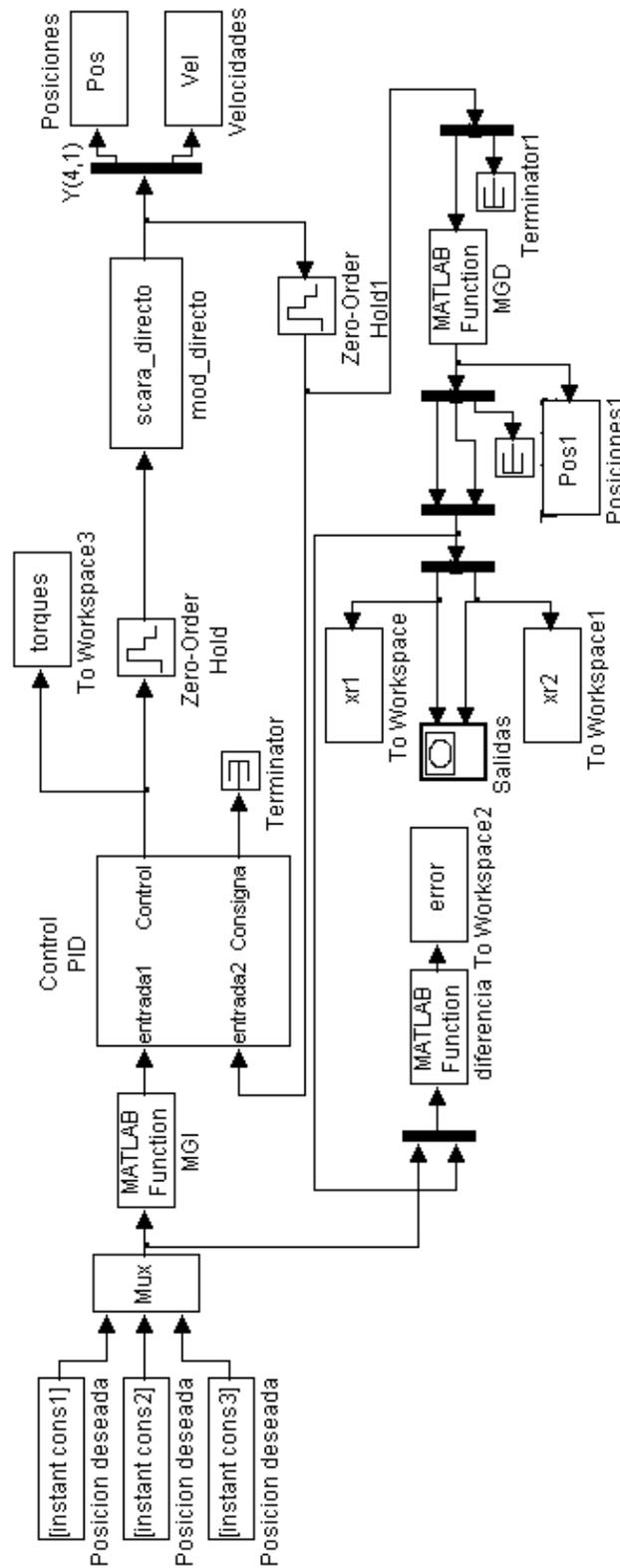


Figura 3.3: Esquema del SCARA con control PID.

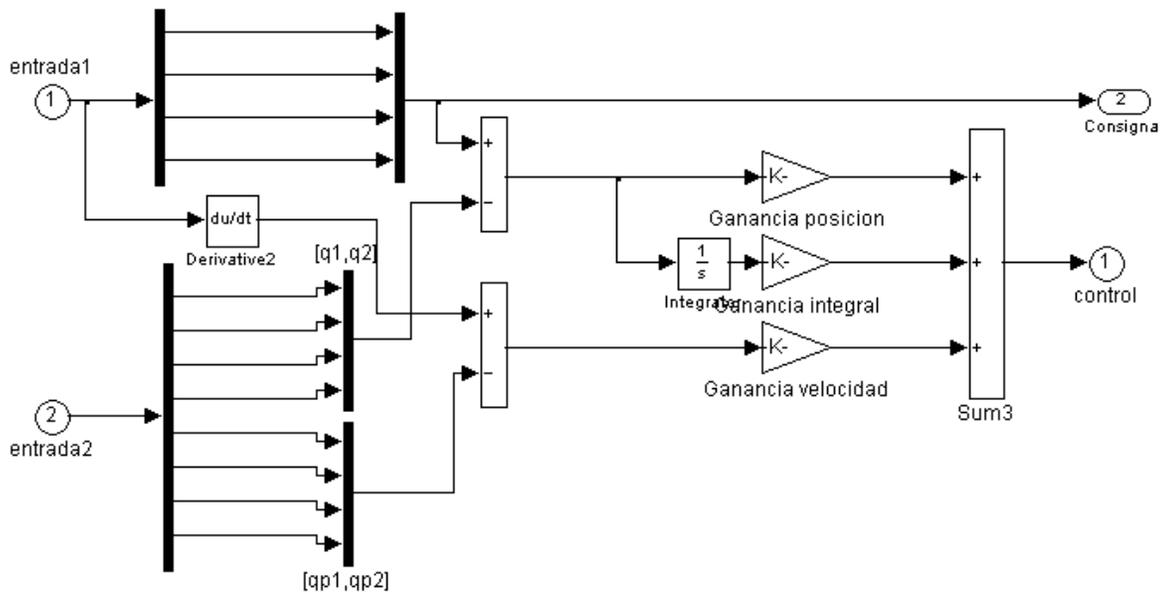


Figura 3.4: Esquema del control PID.

Las figura 3.6 y 3.7 muestran los resultados obtenidos para las trayectorias circular y lineal respectivamente.

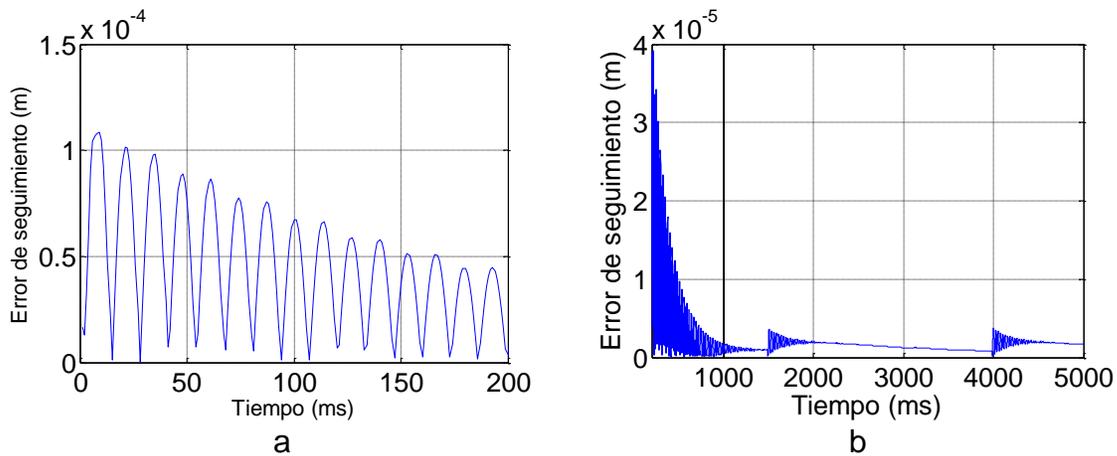


Figura 3.6: Error de seguimiento trayectoria circular control PID.
a) Transitorio. b) Estacionario.

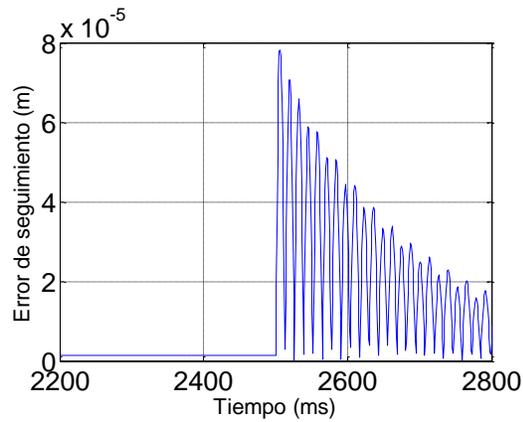


Figura 3.7: Error de seguimiento para el cambio de dirección en trayectoria lineal control PID.

3.3.2 Simulación del control CTC

Las condiciones para la simulación del Control por Par Calculado (CTC), son iguales que para el controlador PID; el esquema del controlador CTC que se desarrolló en *Simulink* se presenta en la figura 3.8. El esquema general del control CTC del robot SCARA es similar al presentado en la figura 3.3, la diferencia está en el bloque de control, el cual es reemplazado por el control CTC.

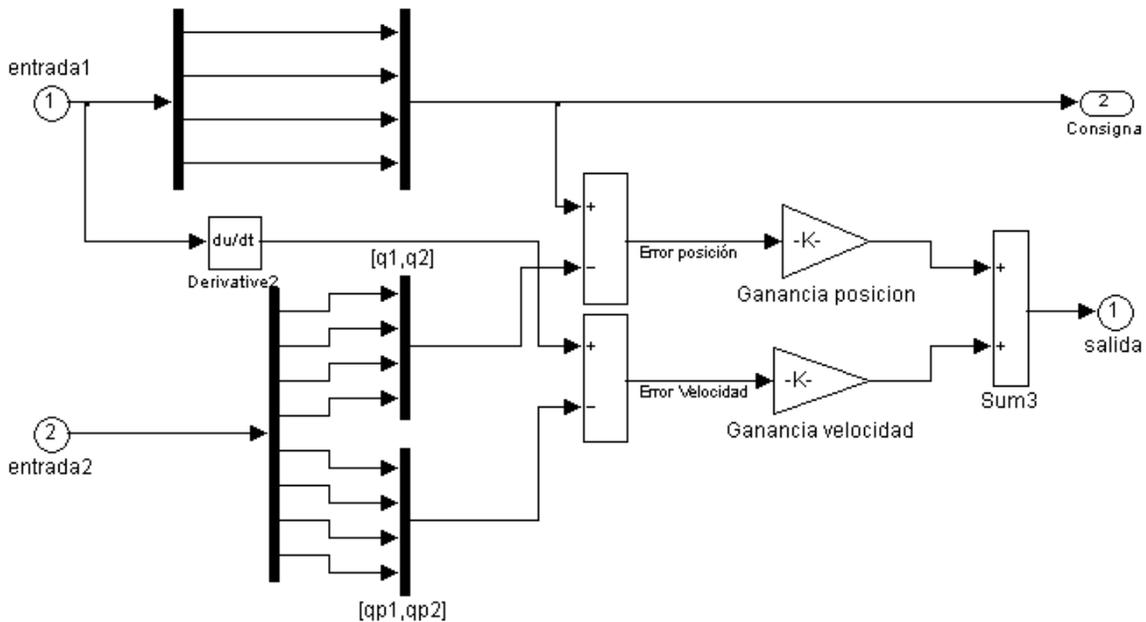


Figura 3.8: Esquema del control CTC.

Las figuras 3.9 y 3.10 presentan los resultados obtenidos del controlador *CTC* con las constantes presentadas en la tabla 3.2.

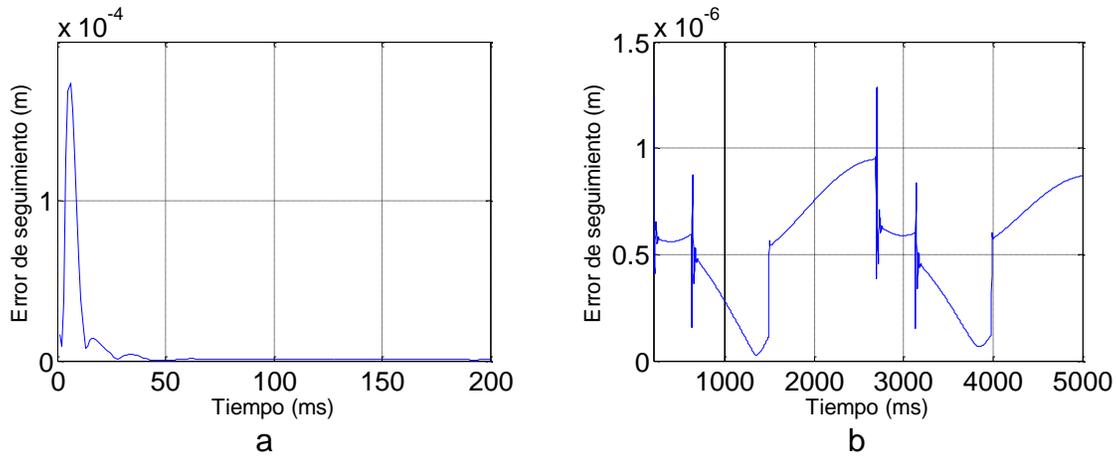


Figura 3.9: Error de seguimiento trayectoria circular control *CTC*. a) Transitorio. b) Estacionario.

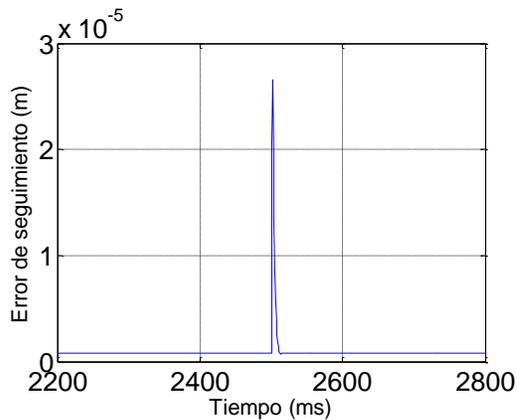


Figura 3.10: Error de seguimiento para el cambio de dirección en trayectoria lineal control *CTC*.

3.3.3 Análisis de resultados

Los resultados en la simulación del control PID y *CTC* del robot *SCARA* teniendo en cuenta el error de seguimiento de las trayectorias circular y lineal se presentan en la tabla 3.3.

Tabla 3.3: Error de seguimiento del control PID y CTC

Error de seguimiento	PID	CTC
Transitorio consigna circular	1.2×10^{-4}	1.8×10^{-4}
Estacionario consigna circular	4×10^{-5}	1.3×10^{-6}
Cambio de dirección consigna lineal	7.8×10^{-5}	2.7×10^{-5}

Es importante resaltar que los resultados de simulación se basan en el modelo nominal del robot, el modelo nominal hace referencia a los modelos obtenidos en el capítulo 2, en los cuales se pueden presentar dinámicas desconocidas y no modeladas que afectan el comportamiento del manipulador; por otra parte el valor de los parámetros se obtiene por medio de procesos de identificación y por tanto no se tienen valores exactos de estos, para el modelo nominal se toma la media de cada uno de los parámetros, el error de cada uno de los parámetros se determina por la desviación estándar (Chan S y Che H., 1999; Zhou K y Doyle J., 1998).

De la tabla 3.8 se puede evidenciar que el control CTC presenta mejor desempeño que el control PID en lo referente al seguimiento de las trayectorias definidas. En el caso de la trayectoria circular la diferencia en estado estacionario es mayor a un orden de magnitud, mientras el control PID presenta un error de seguimiento de 4×10^{-5} metros, el control CTC presenta un error de 1.3×10^{-6} metros. Con respecto al cambio de dirección se evidencia también que el control CTC presenta un mejor desempeño que el control PID.

El control CTC presenta un mejor desempeño que el PID debido a que esta técnica de control realiza un cálculo en línea del modelo dinámico del robot, lo cual le permite generar una mejor acción de control ya que conoce en todo momento el comportamiento dinámico de este, mientras que el control PID desconoce el comportamiento dinámico del robot lo cual hace mas difícil realizar un control.

Capítulo 4

Leyes de control basadas en Lyapunov

El método de Lyapunov es utilizado en la teoría de sistemas no lineales para el análisis de estabilidad; un punto de equilibrio de un sistema se dice estable si todas las soluciones que se inician en las cercanías de ese punto de equilibrio permanecen en las cercanías del mismo, de otro modo el punto de equilibrio es inestable. La teoría de Lyapunov es además una herramienta para el diseño de leyes de control por realimentación, este enfoque busca determinar las condiciones que debe tener la función de Lyapunov para garantizar la estabilidad del sistema que se está analizando (Khalil, 1996; Krstic, *et al.*, 1995).

En este capítulo se consignan las definiciones y fundamentos teórico/matemáticos básicos para el diseño de leyes de control basados en las funciones de Lyapunov, así mismo se encontrarán las bases teóricas para el control *backstepping*, el cual es una técnica recursiva que combina la selección de una función de Lyapunov para el análisis de estabilidad con el control realimentado (Zhang, 2003; Harkegard, 2001).

4.1 Teoría de Lyapunov

La estabilidad es el requerimiento principal en los sistemas de control, la teoría de Lyapunov es una herramienta ampliamente utilizada por la comunidad que estudia e investiga los sistemas de control, para el estudio de la estabilidad de sistemas no lineales y para el diseño de leyes de control. Esta sesión se basa en Krstic y Khalil (Krstic, *et al.*, 1995; Khalil, 1996).

Lema 4.1

Sea $f: [a, b] \times D \rightarrow \mathbb{R}^m$ una función continua, D un dominio de \mathbb{R}^n . Si $\partial f / \partial x$ existe y es continua en $[a, b] \times D$ entonces f es localmente Lipschitz en x en $[a, b] \times D$.

Definición 4.1 (Estabilidad de Lyapunov).

Considere un sistema variante en el tiempo

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t) \quad (4.1)$$

Donde $\mathbf{x} \in \mathfrak{R}^n$ y $\mathbf{f}: D \rightarrow \mathfrak{R}^n$ es localmente Lipschitz desde un dominio $D \subset \mathfrak{R}^n \rightarrow \mathfrak{R}^n$. La solución de (4.1), la cual inicia desde el punto \mathbf{x}_0 en $t_0 \geq 0$, el cual se denota $\mathbf{x}(t; \mathbf{x}_0, t_0)$ con $\mathbf{x}(t; \mathbf{x}_0, t_0) = \mathbf{x}_0$, se puede decir que es:

- *Acotada* sí existe una constante positiva tal que:

$$|\mathbf{x}(t; \mathbf{x}_0, t_0)| < \mathbf{B}(\mathbf{x}_0, t_0), \quad \forall t \geq t_0 \quad (4.2)$$

donde $\mathbf{B}(\mathbf{x}_0, t_0) > 0$.

- *Estable* sí para un $\varepsilon > 0$ existe $\delta(\varepsilon, t_0) > 0$ tal que:

$$|\bar{\mathbf{x}}_0 - \mathbf{x}_0| < \delta \Rightarrow |\mathbf{x}(t; \bar{\mathbf{x}}_0, t_0) - \mathbf{x}(t; \mathbf{x}_0, t_0)| < \varepsilon, \quad \forall t > t_0 \quad (4.3)$$

donde $\bar{\mathbf{x}}_0$ denota la perturbación sobre \mathbf{x}_0 .

- *Asintóticamente estable* sí además de ser estable existe un $r(t_0) > 0$ y para cada $\varepsilon > 0$, un $T(\varepsilon, t_0)$ cumple que:

$$|\bar{\mathbf{x}}_0 - \mathbf{x}_0| < r \Rightarrow |\mathbf{x}(t; \bar{\mathbf{x}}_0, t_0) - \mathbf{x}(t; \mathbf{x}_0, t_0)| < \varepsilon, \quad \forall t > t_0 + T \quad (4.4)$$

- *Globalmente estable* sí (4.4) se satisface para cualquier estado inicial $\mathbf{x}(t_0)$.
- *Inestable*, sí no es estable.

Cuando las constantes B , δ , r y T son independientes de t_0 las propiedades (4.2), (4.3) y (4.4) son uniformes. La estabilidad uniforme es más deseable que la estabilidad por sí sola y más aún es mucho más deseable que un sistema de uniformidad asintóticamente estable, lo cual implica que el sistema debe satisfacer (4.3) y (4.4) con $\delta(\varepsilon, t_0) = \delta(\varepsilon)$, $r(t_0) = r$ y $T(\varepsilon, t_0) = T(\varepsilon)$. La estabilidad asintótica uniforme es una propiedad deseable porque el sistema que la posee presenta un mejor comportamiento frente a disturbios y perturbaciones.

El matemático e ingeniero ruso Lyapunov (Lyapunov, 1992) introdujo la idea de considerar el vector de estados $\mathbf{x}(t)$ dentro de una función escalar $V(\mathbf{x})$ para analizar qué tan cerca del punto de equilibrio se encuentra el sistema. Sí $V(\mathbf{x})$ decrece con el

tiempo entonces el sistema tiende al punto de equilibrio, este análisis se enfoca al estudio de la estabilidad.

Definición 4.2

Dada una función escalar $V(\mathbf{x})$, de la cual se decir que:

- Es definida positiva sí:

$$V(0) = 0 \quad \text{y} \quad V(\mathbf{x}) > 0, \quad \mathbf{x} \neq 0$$

- Es semidefinida positiva sí:

$$V(0) = 0 \quad \text{y} \quad V(\mathbf{x}) \geq 0, \quad \mathbf{x} \neq 0$$

- Es definida negativa sí $-V(\mathbf{x})$ es definida positiva.
- Es semidefinida negativa sí $-V(\mathbf{x})$ es semidefinida positiva.
- Radialmente no acotada sí:

$$V(\mathbf{x}) \rightarrow \infty \quad \text{si} \quad \|\mathbf{x}\| \rightarrow \infty$$

Además de analizar la estabilidad de un sistema no lineal es importante de igual forma estudiar la convergencia a cero del error de seguimiento y una de las herramientas más utilizadas es el teorema de LaSalle y Yoshizawa (Krstic, *et al.*, 1995).

Teorema 4.1 (LaSalle - Yoshizawa)

Sea $\mathbf{x} = \mathbf{0}$ un punto de equilibrio de (4.1) y f es localmente Lipschitz en \mathbf{x} y uniformemente Lipschitz en t . Sea $V: \mathbb{R}^n \rightarrow \mathbb{R}_+$ una función continuamente diferenciable, definida positiva y radialmente no acotada tal que:

$$\frac{\partial V(\mathbf{x})}{\partial \mathbf{x}} f(\mathbf{x}, t) \leq -W(\mathbf{x}) \leq 0, \quad \forall t > 0, \forall \mathbf{x} \in \mathbb{R}^n \quad (4.5)$$

Donde W es una función continua. Entonces toda solución de (4.1) satisface

$$\lim_{t \rightarrow \infty} W(\mathbf{x}(t)) = 0 \quad (4.6)$$

Adicionalmente si $W(\mathbf{x})$ es definida positiva entonces el punto de equilibrio $\mathbf{x} = \mathbf{0}$ es globalmente uniformemente asintóticamente estable (*GUAS: Globally Uniformly Asymptotically Stable*).

Una función $V(\mathbf{x})$ definida positiva que cumple con el teorema 4.1 es llamada función de Lyapunov del sistema. Si esta función se puede encontrar entonces $\mathbf{x} = \mathbf{0}$ es un punto de equilibrio estable. Para analizar si un conjunto invariante sea asintóticamente estable se debe cumplir que la derivada de la función de Lyapunov sea semidefinida negativa.

Corolario 4.1. (Estabilidad asintótica)

Sea $\mathbf{x} = \mathbf{0}$ el único punto de equilibrio de (4.1). Sea $V(\mathbf{x})$ una función escalar definida positiva, continuamente diferenciable y radialmente no limitada tal que:

$$\dot{V}(\mathbf{x}) < 0 \quad \forall \mathbf{x} \in \mathcal{R}^n$$

Sea $\Psi = \{\mathbf{x} \in \mathcal{R}^n \mid \dot{V}(\mathbf{x}) = 0\}$, y se supone que no existe otra solución $\mathbf{x}(t) \equiv \mathbf{0}$ que puede permanecer siempre en Ψ . Entonces el origen es Globalmente Asintóticamente Estable (GAS: *Globally Asymptotically Stable*).

Es claro que el teorema 4.1 y el corolario 4.1 no indican cómo encontrar la función de Lyapunov que los satisfaga, es trabajo de quien analiza el sistema encontrar la función $V(x)$ que demuestre la estabilidad.

4.2 Funciones de control de Lyapunov

Una extensión de la función de Lyapunov es la denominada Función de Control de Lyapunov (*fcl*); el objetivo de las *fcl* es analizar las condiciones que debe cumplir una función de Lyapunov $V(\mathbf{x})$ para garantizar las condiciones deseadas de estabilidad de un sistema.

Se asume para este enfoque un sistema invariante en el tiempo dado por (4.7).

$$\dot{\mathbf{x}} = f(\mathbf{x}, u), \quad \mathbf{x} \in \mathcal{R}^n, \quad u \in \mathcal{R} \quad \text{y} \quad f(0,0) = 0 \quad (4.7)$$

El problema es encontrar una ley de control realimentado ($\sigma(\mathbf{x})$) para la variable de control u , tal que el punto de equilibrio $\mathbf{x} = \mathbf{0}$ del sistema en lazo cerrado determinado por (4.8):

$$\dot{\mathbf{x}} = f(\mathbf{x}, \sigma(\mathbf{x})) \quad (4.8)$$

sea globalmente asintóticamente estable. Para garantizar esta condición de estabilidad, se debe encontrar una función de Lyapunov candidata que cumpla:

$$\dot{\mathbf{x}}(\mathbf{x}) \leq W(\mathbf{x}), \quad W(\mathbf{x}) \in \mathfrak{R} \text{ y } W(\mathbf{x}) < 0 \quad (4.9)$$

y se debe elegir $\sigma(\mathbf{x})$ que para todo $\mathbf{x} \in \mathfrak{R}^n$ se cumpla (4.10).

$$\frac{\partial V}{\partial \mathbf{x}}(\mathbf{x})f(\mathbf{x}, \sigma(\mathbf{x})) \leq -W(\mathbf{x}) \quad (4.10)$$

Esto es difícil de lograr ya que una ley de control que esté de acuerdo con (4.7) puede no satisfacer (4.10) porque se ha seleccionado mal a $V(\mathbf{x})$ y $W(\mathbf{x})$. Para una buena selección de $V(\mathbf{x})$ y $W(\mathbf{x})$ se deben tener mucha experiencia y así lograr que la ley de control diseñada establezca a (4.7) y cumpla con (4.10) simultáneamente.

Definición 4.3.

Sea $V: \mathfrak{R}^n \rightarrow \mathfrak{R}_+$ una función definida positiva suave y radialmente no acotada será una función de control de Lyapunov de (4.7) si:

$$\inf_{u \in \mathfrak{R}} \left\{ \frac{\partial V}{\partial \mathbf{x}}(\mathbf{x})f(\mathbf{x}, u) \right\} < 0, \quad \forall \mathbf{x} \neq \mathbf{0} \quad (4.11)$$

Que (4.11) se cumpla no es necesario pero sí condición suficiente para la existencia de una ley de control que satisfaga (4.10), en otras palabras, la existencia de una *fcl* es equivalente a la estabilidad asintótica global.

Dado el sistema representado por (4.12)

$$\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})u, \quad f(0) = 0 \quad (4.12)$$

y aplicando a este sistema la desigualdad (4.10) se obtiene:

$$\frac{\partial V}{\partial \mathbf{x}}f(\mathbf{x}) + \frac{\partial V}{\partial \mathbf{x}}g(\mathbf{x})\sigma(\mathbf{x}) \leq -W(\mathbf{x}) \quad (4.13)$$

Si $V(\mathbf{x})$ es una *fcl* de (4.12), entonces una ley de control estabilizante $\sigma(\mathbf{x})$ para todo $\mathbf{x} \neq 0$ está dada por la formula de Sontag (Sotang, 1989):

$$u = \sigma(\mathbf{x}) = \begin{cases} -\frac{\frac{\partial V}{\partial \mathbf{x}}f + \sqrt{\left(\frac{\partial V}{\partial \mathbf{x}}f\right)^2 + \left(\frac{\partial V}{\partial \mathbf{x}}g\right)^4}}{\frac{\partial V}{\partial \mathbf{x}}g}, & \frac{\partial V}{\partial \mathbf{x}}g \neq 0 \\ 0, & \frac{\partial V}{\partial \mathbf{x}}g = 0 \end{cases} \quad (4.14)$$

De (4.14) se puede observar que se satisface (4.13) sí:

$$\frac{\partial V}{\partial \mathbf{x}} \mathbf{g}(\mathbf{x}) = 0 \Rightarrow \frac{\partial V}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x}) < 0$$

para este caso entonces:

$$W(\mathbf{x}) = \sqrt{\left(\frac{\partial V}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x})\right)^2 + \left(\frac{\partial V}{\partial \mathbf{x}} \mathbf{g}(\mathbf{x})\right)^4} > 0, \quad \forall \mathbf{x} \neq \mathbf{0}$$

Una caracterización de la ley de control estabilizante $\sigma(x)$ para (4.12) con una *fcl* V es que $\sigma(x)$ es continua en $x = 0$, sí y solo sí la *fcl* satisface que para un $\varepsilon > 0$ hay un $\delta(\varepsilon) > 0$ tal que, sí $x \neq 0$ satisface $|x| < \delta$, entonces existe u con $|u| < \varepsilon$ tal que:

$$\frac{\partial V}{\partial \mathbf{x}} [\mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})] < 0$$

La principal desventaja del concepto de la *fcl* es que para muchos sistemas no lineales una *fcl* es desconocida y encontrarla es tan complicado como el diseño de una ley de control estabilizante.

4.3 Backstepping

Backstepping es un procedimiento recursivo que combina la elección de una función de Lyapunov con el diseño de un control por realimentación (Krstic, *et al*, 1995; Härkegard, 2001; Khalil, 1996). Este procedimiento descompone el problema original en una secuencia de problemas de diseño para sistemas de orden reducido (que hasta pueden llegar a ser escalares). Explotando la flexibilidad adicional que existe con sistemas de bajo orden y escalares, *backstepping* a menudo puede resolver problemas de estabilización, seguimiento y control robusto bajo condiciones menos restrictivas que las encontradas en otros métodos.

4.3.1 Backstepping de un integrador

Backstepping de un integrador es la base para el diseño de leyes de control basadas en este enfoque, a continuación se presenta el procedimiento que demuestra la estabilidad del sistema (4.15) y (4.16) dada una entrada de control u la cual se define en función de una función de Lyapunov V .

Considerando el siguiente sistema:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\xi, \quad \mathbf{f}(\mathbf{0}) = \mathbf{0} \quad (4.15)$$

$$\dot{\xi} = u \quad (4.16)$$

donde $\mathbf{x} \in \mathfrak{R}^n$, $\xi \in \mathfrak{R}$ son los estados y $u \in \mathfrak{R}$ es la entrada de control. Las funciones $\mathbf{f}: D \rightarrow \mathfrak{R}^n$ y $\mathbf{g}: D \rightarrow \mathfrak{R}^n$ son suaves en un dominio $D \subset \mathfrak{R}^n$ que contiene $\mathbf{x} = \mathbf{0}$, y $\mathbf{f}(\mathbf{0}) = \mathbf{0}$.

Se desea diseñar un control por realimentación de estados que establezca el origen $\mathbf{x} = \mathbf{0}$, $\xi = 0$.

El sistema representado por (4.15) y (4.16) se puede analizar como un sistema de dos componentes en cascada como se observa en la figura 4.1.

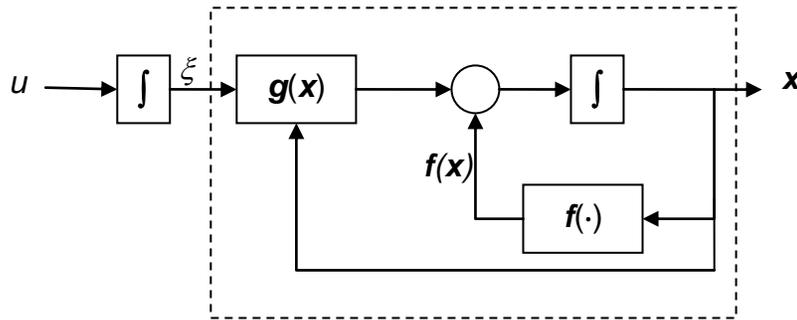


Figura 4.1: Diagrama del sistema (4.15) - (4.16).

Se supone que la primera componente del sistema compuesto por (4.15) puede estabilizarse con una ley de control suave $\xi = \varphi(\mathbf{x})$, con $\varphi(\mathbf{0}) = 0$, es decir que el origen de:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\varphi(\mathbf{x})$$

es asintóticamente estable. Además se supone que se conoce la función de Lyapunov $V(\mathbf{x})$ que satisface:

$$\frac{\partial V}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x}) + \frac{\partial V}{\partial \mathbf{x}} \mathbf{g}(\mathbf{x})\varphi(\mathbf{x}) \leq -W(\mathbf{x}), \quad \forall \mathbf{x} \in D \quad (4.17)$$

donde $W(\mathbf{x})$ es definida positiva. Ahora sumando y restando $\mathbf{g}(\mathbf{x})\varphi(\mathbf{x})$ a (4.15), se obtiene:

$$\dot{\mathbf{x}} = [\mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\varphi(\mathbf{x})] + \mathbf{g}(\mathbf{x})[\xi - \varphi(\mathbf{x})] \quad (4.18)$$

Se realiza un cambio de variable con:

$$z = \xi - \varphi(\mathbf{x}) \quad y \quad v = u - \dot{\varphi} \quad (4.19)$$

Donde:

$$\dot{\varphi} = \frac{\partial \varphi}{\partial \mathbf{x}}(\mathbf{x}) \dot{\mathbf{x}}$$

Sustituyendo (4.15) se obtiene:

$$\dot{\varphi} = \frac{\partial \varphi}{\partial \mathbf{x}} [f(\mathbf{x}) + g(\mathbf{x}) \xi] \quad (4.20)$$

de (4.18) y (4.19) se obtiene:

$$\dot{\mathbf{x}} = [f(\mathbf{x}) + g(\mathbf{x}) \varphi(\mathbf{x})] + g(\mathbf{x}) z \quad (4.21)$$

$$\dot{z} = v \quad (4.22)$$

El sistema compuesto por (4.21) y (4.22) tiene la misma estructura que el sistema inicial, solo que ahora la primera componente del sistema (ecuación (4.21)) tiene un punto de equilibrio asintóticamente estable en el origen cuando su entrada z es cero. Esta característica va a ser explotada en el diseño de un control v que establezca todo el sistema.

La figura 4.2 muestra el diagrama en bloques del sistema (4.21) - (4.22).

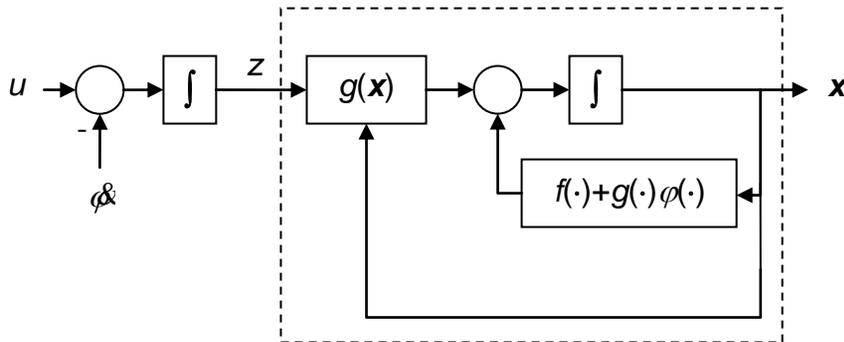


Figura 4.2: Diagrama del sistema (4.21) - (4.22).

Considerando como función de Lyapunov candidata de (4.21) - (4.22) a:

$$V_a(x, z) = V(\mathbf{x}) + \frac{1}{2} z^2 \quad (4.23)$$

cuya derivada satisface:

$$\dot{V}_a = \frac{\partial V}{\partial \mathbf{x}} [f(\mathbf{x}) + g(\mathbf{x})\varphi(\mathbf{x})] + \frac{\partial V}{\partial \mathbf{x}} g(\mathbf{x})z + zv \leq -W(\mathbf{x}) + \left[\frac{\partial V}{\partial \mathbf{x}} g(\mathbf{x}) + v \right] z$$

Eligiendo v como:

$$v = -\frac{\partial V}{\partial \mathbf{x}} g(\mathbf{x}) - kz, \quad k > 0 \quad (4.24)$$

se obtiene:

$$\dot{V}_a \leq -W(\mathbf{x}) - kz^2$$

que muestra que el origen $x = 0, z = 0$ es asintóticamente estable. Como $\varphi(0) = 0$ se concluye que el origen $x = 0, \xi = 0$ es asintóticamente estable. Procediendo aritméticamente entre (4.19), (4.20) y (4.24) se tiene:

$$u = \frac{\partial \varphi}{\partial \mathbf{x}} [f(\mathbf{x}) + g(\mathbf{x})\xi] - \frac{\partial V}{\partial \mathbf{x}} g(\mathbf{x}) - k[\xi - \varphi(\mathbf{x})] \quad (4.25)$$

Sí las hipótesis son validadas globalmente y $V(\mathbf{x})$ es radialmente no acotada, se concluye que el origen es globalmente asintóticamente estable. Esta técnica se denomina *integrador backstepping* ya que al control virtual $\xi = \varphi(\mathbf{x})$ se le ubica un integrador atrás para poder obtener la ley de control real u .

Esta técnica se resume en el siguiente lema:

Lema 4.2. (Integrador *backstepping*)

Sea el sistema (4.15) - (4.16). Sea $\varphi(\mathbf{x})$ una ley de control por realimentación de estados estabilizante de (4.15) con $\varphi(\mathbf{0}) = 0$ y sea $V(\mathbf{x})$ una función de Lyapunov que satisface (4.17) con una función definida positiva $W(\mathbf{x})$. Entonces el control por realimentación de estados (4.25) estabiliza el origen del sistema (4.15) - (4.16) con la función de Lyapunov:

$$V_a(\mathbf{x}, z) = V(\mathbf{x}) + \frac{1}{2} [\xi - \varphi(\mathbf{x})]^2$$

Además, si todas las hipótesis valen globalmente y V es radialmente no acotada, el origen será globalmente asintóticamente estable.

4.3.2 *Backstepping* de sistemas en realimentación estricta

De manera más general, *backstepping* se puede aplicar a sistemas tales como:

$$\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})\xi \quad (4.26)$$

$$\dot{\xi} = f_a(\mathbf{x}, \xi) + g_a(\mathbf{x}, \xi)u \quad (4.27)$$

donde f_a y g_a son suaves y $g_a(\mathbf{x}, \xi) \neq 0$ en el dominio de interés; se puede notar que (4.27) no es un integrador puro sino un integrador perturbado por la presencia de las no linealidades f_a y g_a . Esta perturbación es manejable, sin embargo mediante la transformación (4.28):

$$u = \frac{1}{g_a(\mathbf{x}, \xi)} [u_a - f_a(\mathbf{x}, \xi)] \quad (4.28)$$

(4.27) se reduce a un integrador puro $\dot{\xi} = u_a$.

Si conocemos como antes el par de funciones $\varphi(\mathbf{x})$ y $V(\mathbf{x})$ correspondientes a la estabilización de (4.26), podemos tomar u_a igual al control (4.25) y combinando con (4.28) se puede obtener el control por *backstepping* para (4.26) - (4.27), la cual se indica en (4.29):

$$u = \varphi_a(\mathbf{x}, \xi) A \frac{1}{g_a(\mathbf{x}, \xi)} \left\{ \frac{\partial \varphi}{\partial \mathbf{x}} [f(\mathbf{x}) + g(\mathbf{x})\xi] - \frac{\partial V}{\partial \mathbf{x}} g(\mathbf{x}) - k[\xi - \varphi(\mathbf{x})] - f_a(\mathbf{x}, \xi) \right\} \quad (4.29)$$

con $k > 0$. La función de Lyapunov total es:

$$V_a(\mathbf{x}, \xi) = V(\mathbf{x}) + \frac{1}{2} [\xi - \varphi(\mathbf{x})]^2 \quad (4.30)$$

Aplicando recursivamente el control por *backstepping* de un integrador, se puede estabilizar el sistema en forma de realimentación estricta, así:

$$\begin{aligned}\dot{\mathbf{x}} &= f_0(\mathbf{x}) + g_0(\mathbf{x})\xi_1 \\ \dot{\xi}_1 &= f_1(\mathbf{x}, \xi_1) + g_1(\mathbf{x}, \xi_1)\xi_2 \\ \dot{\xi}_2 &= f_2(\mathbf{x}, \xi_1, \xi_2) + g_2(\mathbf{x}, \xi_1, \xi_2)\xi_3 \\ &\vdots \\ \dot{\xi}_n &= f_n(\mathbf{x}, \xi_1, \dots, \xi_n) + g_n(\mathbf{x}, \xi_1, \dots, \xi_n)u\end{aligned}$$

donde $\mathbf{x} \in \mathbb{R}^m$, $u, \xi_1, \dots, \xi_n \in \mathbb{R}$ y donde las funciones f_0, \dots, f_n se anulan en el origen, se asume además que $g_i(\mathbf{x}, \xi_1, \dots, \xi_i) \neq 0$ para $i=1, \dots, n$. La razón por la cual se llaman sistemas en realimentación estricta es porque las funciones f_i y g_i en la n ecuaciones de $\dot{\xi}_i$ depende solo de $\mathbf{x}, \xi_1, \dots, \xi_i$, es decir, sólo de variables que son realimentadas.

El procedimiento recursivo de *backstepping* comienza, como se explicó anteriormente, con el sistema:

$$\dot{\mathbf{x}} = f_0(\mathbf{x}) + g_0(\mathbf{x})\xi_1 \quad (4.31)$$

donde ξ_1 es la entrada de control virtual. Se asume que se pueden encontrar una ley de control $\xi_1 = \varphi_0(\mathbf{x})$ con $\varphi_0(\mathbf{0}) = 0$ y una función de Lyapunov $V_0(\mathbf{x})$ tal que cumplan con (4.17), en el dominio de interés para alguna función $W_0(\mathbf{x})$ definida positiva.

El siguiente paso es considerar el sistema:

$$\begin{aligned}\dot{\mathbf{x}} &= f_0(\mathbf{x}) + g_0(\mathbf{x})\xi_1 \\ \dot{\xi}_1 &= f_1(\mathbf{x}, \xi_1) + g_1(\mathbf{x}, \xi_1)\xi_2\end{aligned}$$

el cual es la misma representación que presenta (4.26) - (4.27). Aplicando (4.29) y (4.30), se tiene:

$$\begin{aligned}\varphi(\mathbf{x}, \xi_1) &= \frac{1}{g_1} \left[\frac{\partial \varphi_0}{\partial \mathbf{x}} (f_0 + g_0 \xi_1) - \frac{\partial V_0}{\partial \mathbf{x}} g_0 - k_1 (\xi_1 - \varphi_0) - f_1 \right], \quad k_1 > 0 \\ V_1(\mathbf{x}, \xi_1) &= V_0 + \frac{1}{2} [\xi_1 - \varphi_0]^2\end{aligned}$$

donde $\varphi(\mathbf{x}, \xi_1)$ es el control estabilizante y $V_1(\mathbf{x}, \xi_1)$ es la correspondiente función de Lyapunov para este sistema. Luego se considera el sistema:

$$\begin{aligned}\dot{\mathbf{x}} &= f_0(\mathbf{x}) + g_0(\mathbf{x})\xi_1 \\ \dot{\xi}_1 &= f_1(\mathbf{x}, \xi_1) + g_1(\mathbf{x}, \xi_1)\xi_2 \\ \dot{\xi}_2 &= f_2(\mathbf{x}, \xi_1, \xi_2) + g_2(\mathbf{x}, \xi_1, \xi_2)\xi_3\end{aligned}$$

y como un caso particular de (4.26) - (4.27) para:

$$\mathbf{x} = \begin{bmatrix} \mathbf{x} \\ \xi_1 \end{bmatrix}, \quad \xi = \xi_2, \quad f = \begin{bmatrix} f_0 + g_0\xi_1 \\ f_1 \end{bmatrix}, \quad g = \begin{bmatrix} 0 \\ g_1 \end{bmatrix}, \quad f_a = f_2, \quad g_a = g_2$$

Aplicando nuevamente (4.29) y (4.30):

$$\begin{aligned}\varphi(\mathbf{x}, \xi_1, \xi_2) &= \frac{1}{g_2} \left[\frac{\partial \varphi_1}{\partial \mathbf{x}} (f_0 + g_0\xi_1) + \frac{\partial \varphi_1}{\partial \xi_1} (f_1 + g_1\xi_2) - \frac{\partial V_1}{\partial \xi_1} g_1 - k_2(\xi_2 - \varphi_1) - f_2 \right], \quad k_2 > 0 \\ V_2(\mathbf{x}, \xi_1, \xi_2) &= V_1 + \frac{1}{2} [\xi_2 - \varphi_1]^2\end{aligned}$$

se repite este procedimiento n veces hasta obtener la función de control estabilizante u y la función de Lyapunov V_n del sistema completo.

4.3.3 Diseño de fc por amortiguamiento no lineal

La potencialidad del control por *backstepping* se evidencia en la presencia de incertidumbres y parámetros desconocidos del sistema, en esta sesión se estudiará lo relacionado con el amortiguamiento no lineal (*nonlinear damping*) y el diseño de controladores por *backstepping* con incertidumbre, los temas de esta sesión son extractados de (Krstic, *et al.*, 1995).

4.3.3.1 Amortiguamiento no lineal

El método de diseño de las fc por amortiguamiento no lineal permite robustecer una ley de control dada, tal que tolere ciertas incertidumbres y errores de modelado que satisfacen la *condición de apareamiento (matching condition)*, es decir, incertidumbres y errores que ocurren en el mismo punto del lazo donde se aplica la entrada de control (Khalil, 1996).

Definición 4.4. (Estabilidad entrada a estado (*ISS: Input to State Stability*))

Dado el sistema (4.7) se dice que es *ISS* si para $\mathbf{x}(0)$ y para cada entrada $u(\cdot)$ continua y acotada sobre $[0, \infty)$, existe una solución para todo $t \geq 0$ y satisface:

$$|\mathbf{x}| \leq \beta(\mathbf{x}(0), t) + \gamma\left(\sup_{0 \leq \tau \leq t} |u(\tau)|\right), \quad \forall t \geq 0$$

donde $\beta(s, t)$ y $\gamma(s)$ son funciones estrictamente crecientes de $s \in \mathfrak{R}_+$ con $\beta(0, t) = 0$, $\gamma(0) = 0$, mientras que β es una función decreciente de t con:

$$\lim_{t \rightarrow \infty} \beta(s, t) = 0, \quad \forall s \in \mathfrak{R}_+$$

Definición 4.5.

Una función continua $\gamma: [0, a) \rightarrow \mathfrak{R}_+$; se dice que es una función que clase \mathbf{K} si es estrictamente creciente y $\gamma(0) = 0$ se dice que pertenece a la clase \mathbf{K}_∞ , si $a = \infty$ y $\gamma(r) \rightarrow \infty$ cuando $r \rightarrow \infty$.

Lema 4.3. (Amortiguamiento no lineal)

Considerando el sistema perturbado:

$$\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})[u + \Gamma(\mathbf{x})\Delta(\mathbf{x}, u, t)] \quad (4.32)$$

donde $\Gamma(\mathbf{x})$ es un vector ($p \times 1$) de funciones no lineales suaves conocidas, y $\Delta(t, \mathbf{x}, u)$ es un vector ($1 \times p$) de incertidumbres no lineales, y ambos son uniformemente acotados para todo t, \mathbf{x}, u .

Si (4.13) se satisface con una $W(\mathbf{x})$ definida positiva y radialmente no acotada, entonces la ley de control:

$$u = \varphi(\mathbf{x}) - k \frac{\partial V}{\partial \mathbf{x}}(\mathbf{x}) g(\mathbf{x}) |\Gamma(\mathbf{x})|^2, \quad k > 0 \quad (4.33)$$

cuando es aplicada a (4.32) hace que el sistema sea *ISS* con respecto a la entrada de disturbio $\Delta(t, \mathbf{x}, u)$ y garantiza que \mathbf{x} sea uniformemente acotada global y converja al conjunto:

$$R = \left\{ \mathbf{x} : |\mathbf{x}| \leq \gamma_1^{-1} \circ \gamma_2 \circ \gamma_3^{-1} \left(\frac{\|\Delta\|_\infty^2}{4k} \right) \right\} \quad (4.34)$$

donde $\gamma_1, \gamma_2, \gamma_3$ son funciones clase \mathbf{K}_∞ tal que:

$$\gamma_1(|\mathbf{x}|) \leq V(\mathbf{x}) \leq \gamma_2(|\mathbf{x}|) \quad (4.35)$$

$$\gamma_3(|\mathbf{x}|) \leq W(\mathbf{x}) \quad (4.36)$$

La demostración está detallada en (Krstic, 95).

4.3.4 Backstepping con incertidumbre

El lema 4.3 hace referencia al caso en el cual la incertidumbre cumple con la condición de *matching*. Teniendo en cuenta el lema 4.3 y el lema 4.2 y combinándolos se extiende más allá de la condición de *matching*.

Lema 4.4.

Considerando el sistema:

$$\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x}) + \mathbf{F}(\mathbf{x})\Delta_1(\mathbf{x}, u, t) \quad (4.37)$$

donde $\mathbf{x} \in \mathfrak{R}^n$, $u \in \mathfrak{R}$, $\mathbf{F}(\mathbf{x})$ es una matriz $n \times q$ de funciones suaves no lineales conocidas y Δ_1 es un vector de incertidumbres no lineales, el cual es uniformemente acotado para todo valor de \mathbf{x} , u , t . Se supone que existe una ley de control $u = \alpha(\mathbf{x})$ que hace a $\mathbf{x}(t)$ globalmente uniformemente acotada, y que es estabilizada por funciones radialmente acotadas $V(\mathbf{x})$, $W(\mathbf{x})$ y una constante b tal que:

$$\frac{\partial V}{\partial \mathbf{x}}(\mathbf{x}) [f(\mathbf{x}) + g(\mathbf{x})\alpha(\mathbf{x}) + \mathbf{F}(\mathbf{x})\Delta_1(\mathbf{x})] \leq -W(\mathbf{x}) + b \quad (4.38)$$

Ahora considerando el sistema aumentado:

$$\begin{aligned} \dot{\mathbf{x}} &= f(\mathbf{x}) + g(\mathbf{x})\xi + \mathbf{F}(\mathbf{x})\Delta_1(\mathbf{x}, u, t) \\ \dot{\xi} &= u + \boldsymbol{\varphi}(\mathbf{x}, \xi)^T \Delta_2(\mathbf{x}, \xi, u, t) \end{aligned} \quad (4.39)$$

donde $\boldsymbol{\varphi}$ es un vector de funciones suaves no lineales conocidas y Δ_2 es un vector de incertidumbres no lineales las cuales son uniformemente acotadas para todo valor \mathbf{x} , ξ , u , t . Para este sistema la ley de control:

$$\begin{aligned} u = & -c[\xi - \alpha(\mathbf{x})] + \frac{\partial \alpha}{\partial \mathbf{x}}(\mathbf{x}) [f(\mathbf{x}) + g(\mathbf{x})\xi] - \frac{\partial V}{\partial \mathbf{x}}(\mathbf{x})g(\mathbf{x}) \\ & - k[\xi - \alpha(\mathbf{x})] \left\{ \left| \boldsymbol{\varphi}(\mathbf{x}, \xi) \right|^2 + \left| \frac{\partial \alpha}{\partial \mathbf{x}}(\mathbf{x})\mathbf{F}(\mathbf{x}) \right|^2 \right\} \end{aligned} \quad (4.40)$$

garantiza que \mathbf{x} y ξ serán global uniformemente acotadas con algún $c > 0$ y $k > 0$ (ver demostración en (Krstic, 95)).

Capítulo 5

Diseño de leyes de control robustas

En los capítulos anteriores se estudió los modelos del robot SCARA, se realizó el control PID y CTC y se desarrolló una simulación de ambas estrategias de control para estudiar el comportamiento del robot en el seguimiento de dos trayectorias; ahora basados en la teoría de Lyapunov se desarrollaran dos leyes de control, la primera basada en Spong (Spong, 1992), la cual se ha convertido en el punto de partida para el desarrollo de estrategias de control adaptativo y óptimo (Burkan y Uzmay, 2003; Mnif, 2004; Alonge, *et. al.*, 2004; Burkan y Uzmay, 2005; Lin y Brandt, 1998), las cuales tienen en cuenta la incertidumbre paramétrica, y la segunda ley de control, que representa el aporte de este trabajo, radica en la apropiación de la estrategia de control *backstepping* teniendo en cuenta la incertidumbre paramétrica y aplicarla a un robot manipulador, aunque esta técnica ya a sido aplicada a manipuladores de los cuales se conoce la dinámica de los actuadores (Chun *et al.* 1996, Chun *et al.* 1997; Ishii, 1999), en este trabajo esta dinámica no sea tenido en cuenta, debido a que no se cuenta con el robot real para realizar un proceso de identificación de los actuadores, mas aun este proceso de identificación no es sencillo.

Las estrategias de control robusto son desarrolladas con el fin de estabilizar sistemas que presentan errores en el modelado y dinámicas no modeladas. Los modelos con incertidumbre son frecuentemente utilizados en robótica, donde dichas incertidumbres se deben a cambios en la carga del robot, simplificación del modelo dinámico, fuerzas de fricción, entre otras. La incertidumbre ocasiona que los parámetros de desempeño del sistema, como el error de seguimiento, se vean afectados (Sage, *et al.*, 1999), por tal razón es importante el diseño de esquemas de control robusto.

En este capítulo se realizará el diseño de las dos leyes de control mencionadas anteriormente.

5.1 Ley de control robusta basada en Lyapunov

El enfoque clásico para el análisis de estabilidad de sistemas no lineales se conoce como el método directo de Lyapunov. Frecuentemente en el control de robots se aplica el método inverso el cual consiste en definir una ley de control *a priori* y luego por medio del análisis de estabilidad de Lyapunov se prueba que el sistema con la ley de control seleccionada cumple con un tipo de estabilidad (Sage, *et al.*, 1999).

En esta sesión se desarrollará la ley de control propuesta por Spong (Spong, 1992) y Burlan y Uzmay (Burkan y Uzmay, 2005), aplicada al robot SCARA de 4 grados de libertad

A continuación se listan algunas propiedades requeridas en este capítulo para el diseño de las leyes de control.

Propiedad 5.1: (Khalil y Dembre, 2002).

Para todo $\mathbf{q} \in \mathfrak{R}^n$ la matriz $\mathbf{M}(\mathbf{q})$ es simétrica y definida positiva.

Propiedad 5.2: (Khalil y Dembre, 2002; Mnif, 2004).

Dada la definición de la matriz \mathbf{C} (ecuación (2.23)) hace que

$$\dot{\mathbf{M}}(\mathbf{q}) - 2\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \quad (5.1)$$

sea antisimétrica, luego

$$\mathbf{x}^T [\dot{\mathbf{M}}(\mathbf{q}) - 2\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})] \mathbf{x} = 0 \quad \forall \mathbf{x} \in \mathfrak{R}^n \quad (5.2)$$

Propiedad 5.3: (Propiedades de normas de matrices)(Chen, 1999).

Sean \mathbf{B} y \mathbf{D} matrices y \mathbf{x} un vector, cuyas dimensiones permitan la validez de las operaciones mostradas a continuación, luego se cumple que:

$$\begin{aligned} \|\mathbf{B}\mathbf{x}\| &\leq \|\mathbf{B}\| \|\mathbf{x}\| \\ \|\mathbf{B} + \mathbf{D}\| &\leq \|\mathbf{B}\| + \|\mathbf{D}\| \\ \|\mathbf{B}\mathbf{D}\| &\leq \|\mathbf{B}\| \|\mathbf{D}\| \end{aligned}$$

5.1.1 Ley de control robusta de Spong (Spong, 1992).

El modelo dinámico del robot, sin tener en cuenta las fricciones y las señales de disturbio se presenta en la ecuación (5.3).

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau \quad (5.3)$$

Otra forma de representación del modelo dinámico se presenta en la ecuación (5.4).

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = Y(q, \dot{q}, \ddot{q})\pi \quad (5.4)$$

Donde π es un vector constante n -dimensional de los parámetros del robot y Y es una matriz $m \times n$, la cual es función de la posición, velocidad y aceleración de las articulaciones del robot.

Tomando el modelo dinámico del SCARA (Ecuación (2.26)) y expresándolo como (5.4) se obtiene:

$$\pi = [XX_j \quad XY_j \quad XZ_j \quad YY_j \quad YZ_j \quad ZZ_j \quad MX_j \quad MY_j \quad MZ_j \quad M_j \quad Ia_j]^T \quad (5.5)$$

y

$$Y = \begin{bmatrix} y_{11} & y_{12} & y_{13} & y_{14} & y_{15} & y_{16} & y_{17} & y_{18} & y_{19} & y_{110} & y_{111} \\ y_{21} & y_{22} & y_{23} & y_{24} & y_{25} & y_{26} & y_{27} & y_{28} & y_{29} & y_{210} & y_{211} \\ y_{31} & y_{32} & y_{33} & y_{34} & y_{35} & y_{36} & y_{37} & y_{38} & y_{39} & y_{310} & y_{311} \\ y_{41} & y_{42} & y_{43} & y_{44} & y_{45} & y_{46} & y_{47} & y_{48} & y_{49} & y_{410} & y_{411} \end{bmatrix} \quad (5.6)$$

Donde

$$\begin{aligned} y_{11} &= \ddot{q}_1; & y_{12} &= \ddot{q}_1 + 2\ddot{q}_2; & y_{13} &= \ddot{q}_1 + 2\ddot{q}_2 + \ddot{q}_3 \\ y_{14} &= \ddot{q}_1 + \ddot{q}_2 + \ddot{q}_3; & y_{15} &= 2C_2d_2(\ddot{q}_1 + \ddot{q}_2) \\ y_{16} &= (d_3C_3 + d_2C_3C_2 - d_2S_3S_2)(2\ddot{q}_1 + 2\ddot{q}_2 + \ddot{q}_3) + d_3C_3\ddot{q}_2 \\ y_{17} &= 2d_2S_2(\ddot{q}_1 - \ddot{q}_2) \\ y_{18} &= -(d_3S_3 + d_2S_3C_2 + d_2C_3S_2)(2\ddot{q}_1 + 2\ddot{q}_2 + \ddot{q}_3) - d_3S_3\ddot{q}_2 \\ y_{19} &= y_{110} = y_{111} = y_{21} = 0; & y_{22} &= 2\ddot{q}_1 + \ddot{q}_2; & y_{23} &= 2\ddot{q}_1 + \ddot{q}_2 + \ddot{q}_3 \\ y_{24} &= \ddot{q}_1 + \ddot{q}_2 + \ddot{q}_3; & y_{25} &= 2C_2d_2\ddot{q}_1 \\ y_{26} &= 2(d_3C_3 + d_2C_3C_2 - d_2S_3S_2 + d_3C_3)\ddot{q}_1 + 2d_3C_3\ddot{q}_2 + d_3C_3\ddot{q}_3 \\ y_{27} &= -2d_2S_2\ddot{q}_1 \\ y_{28} &= -2(d_3S_3 + d_2S_3C_2 + d_2C_3S_2 + d_3S_3)\ddot{q}_1 - 2d_3S_3\ddot{q}_2 - 2d_3S_3\ddot{q}_3 \\ y_{29} &= \ddot{q}_2; & y_{210} &= y_{211} = 0; & y_{31} &= y_{32} = 0 \end{aligned}$$

$$\begin{aligned}
y_{33} = y_{34} &= \ddot{q}_{r1} + \ddot{q}_{r2} + \ddot{q}_{r3}; y_{35} = 0; \\
y_{36} &= (d_3 C3 + d_2 C3C2 - d_2 S3S2 + d_3 C3) \ddot{q}_{r1} + d_3 C3 \ddot{q}_{r2}; y_{37} = 0; \\
y_{38} &= -2(d_3 S3 + d_2 S3C2 + d_2 C3S2) \ddot{q}_{r1} - 2d_3 S3 \ddot{q}_{r2}; y_{39} = 0; y_{310} = \ddot{q}_{r3} \\
y_{311} &= 0; y_{41} = y_{42} = y_{43} = y_{44} = y_{45} = y_{46} = y_{47} = y_{48} = y_{49} = y_{410} = 0 \\
y_{411} &= \ddot{q}_{r4}
\end{aligned}$$

Para una trayectoria específica los vectores de posición (\mathbf{q}_d), velocidad ($\dot{\mathbf{q}}_d$) y aceleración ($\ddot{\mathbf{q}}_d$) deseados son conocidos; en las ecuaciones (5.7) y (5.8) se definen los vectores de error de posición y velocidad respectivamente.

$$\mathbf{q}^e = \mathbf{q} - \mathbf{q}_d \quad (5.7)$$

$$\dot{\mathbf{q}}^e = \dot{\mathbf{q}} - \dot{\mathbf{q}}_d \quad (5.8)$$

Se define ahora una corrección de los vectores de velocidad y aceleración deseados debido a los efectos de las no linealidades y acoplamiento de las ecuaciones del modelo del robot. Estos vectores se definen así:

$$\ddot{\mathbf{q}}^c = \ddot{\mathbf{q}}_d - \Lambda \dot{\mathbf{q}}^e \quad (5.9)$$

$$\ddot{\mathbf{q}}^r = \ddot{\mathbf{q}}_d - \Lambda \dot{\mathbf{q}}^e \quad (5.10)$$

Donde Λ es una matriz diagonal y definida positiva.

Se considera a continuación una ley de control para el sistema nominal dada por la ecuación (5.11).

$$\begin{aligned}
\tau_0 &= \mathbf{M}_0(\mathbf{q}) \ddot{\mathbf{q}}^c + \mathbf{C}_0(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}}^c + \mathbf{G}_0(\mathbf{q}) - \mathbf{K}\sigma \\
&= \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \pi_0 - \mathbf{K}\sigma
\end{aligned} \quad (5.11)$$

Donde π_0 es el vector de parámetros nominales del modelo dinámico del robot, los cuales se consignan en la tabla 2.5 y $\mathbf{K}\sigma$ es el vector de la acción derivativa de la ley de control, donde σ representa la corrección del error de velocidad definida como:

$$\sigma = \dot{\mathbf{q}} - \dot{\mathbf{q}}_d = \dot{\mathbf{q}}^e + \Lambda \mathbf{q}^e \quad (5.12)$$

Debido a que los modelos obtenidos del robot *SCARA* están sujetos a dinámicas desconocidas o no modeladas y además se presenta una variación entre los valores nominales del robot y los reales, existe una diferencia entre el modelo matemático

encontrado del robot y la realidad. A este error se le denomina incertidumbre (Kemin y Doyle, 1998); para el caso del robot SCARA se asume una incertidumbre paramétrica representada por ρ , el cual representa la máxima variación entre los parámetros nominales del robot y los reales y se obtiene según la ecuación (5.13).

$$\|\pi\| = \|\pi - \pi_0\| \leq \rho \quad (5.13)$$

ρ es desconocida; ahora se tendrá en cuenta esta incertidumbre para el diseño de la ley de control robusta (τ) en términos de la ley de control nominal así:

$$\tau = \tau_0 + Y(q, \dot{q}, \ddot{q}, \ddot{q}_r) \mu(t) \quad (5.14)$$

sustituyendo (5.11) en (5.14) se tiene:

$$\tau = Y(q, \dot{q}, \ddot{q}, \ddot{q}_r)(\pi_0 + \mu) - K\sigma \quad (5.15)$$

Donde $\mu(t)$ es la ley de control adicional que debe proveer al sistema robustez a la incertidumbre paramétrica representada por π . Sustituyendo (5.15) en (5.3) se tiene:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + K\sigma = Y(q, \dot{q}, \ddot{q}, \ddot{q}_r)(\pi_0 + \mu) \quad (5.16)$$

Teniendo en cuenta la ecuación (5.12), se puede observar que el vector σ es una combinación lineal del vector de error de posición y el error de velocidad. Luego un punto de equilibrio estable de (5.16) es $\dot{q} = 0$ y $q = 0$, esto es equivalente a decir que el punto de equilibrio es $\dot{q} = 0$ y $\sigma = 0$.

La función de Lyapunov candidata para (5.16) es (Spong 1992):

$$V = \frac{1}{2} \sigma^T M \sigma + \dot{q}^T \Lambda^T K \dot{q} \quad (5.17)$$

Donde Λ es una matriz diagonal y definida positiva, además de la propiedad 5.1 se tiene que V es definida positiva. Con el fin de cumplir con el corolario 4.1 se procede a derivar V para luego establecer $\dot{V} < 0$.

$$\dot{V} = \frac{1}{2} \dot{\sigma}^T M \sigma + \sigma^T M \dot{\sigma} + \dot{q}^T \Lambda^T K \dot{q} \quad (5.18)$$

Despejando $M\dot{\sigma}$ de (5.16) y sustituyendo en (5.18) se tiene:

$$\dot{V} = \frac{1}{2} \dot{\sigma}^T M \sigma + \sigma^T (Y(\pi_0 + \mu) - K\sigma - C\sigma) + \dot{q}^T \Lambda^T K \dot{q} \quad (5.19)$$

Procediendo aritméticamente:

$$\begin{aligned} V^* &= \frac{1}{2} \sigma^T M \sigma + \sigma^T Y (\hat{\theta} + \mu) - \sigma^T K \sigma - \sigma^T C \sigma + \hat{\varphi}^T \Lambda^T K \hat{\varphi} \\ &= \frac{1}{2} \sigma^T (M - 2C) \sigma + \sigma^T Y (\hat{\theta} + \mu) - \sigma^T K \sigma + \hat{\varphi}^T \Lambda^T K \hat{\varphi} \end{aligned} \quad (5.20)$$

Teniendo en cuenta la propiedad 5.2:

$$V^* = \sigma^T Y (\hat{\theta} + \mu) - \sigma^T K \sigma + \hat{\varphi}^T \Lambda^T K \hat{\varphi} \quad (5.21)$$

Sustituyendo (5.12) en (5.21):

$$\begin{aligned} V^* &= \sigma^T Y (\hat{\theta} + \mu) - \hat{\varphi}^T K \hat{\varphi} - \hat{\varphi}^T \Lambda^T K \Lambda \hat{\varphi} \\ &= -x^T Q x + \sigma^T Y (\hat{\theta} + \mu) \end{aligned} \quad (5.22)$$

Donde $x^T = [\hat{\varphi}^T, \hat{\varphi}^T]$ y $Q = \text{Diag}\{\Lambda^T K \Lambda, K\}$.

Con el fin de garantizar que V^* sea definida negativa y en consecuencia garantizar la estabilidad asintótica, se define la ley de control robusta como:

$$\mu = \begin{cases} -\rho \frac{Y^T \sigma}{\|Y^T \sigma\|} & \text{sí } \|Y^T \sigma\| > \varepsilon \\ -\rho \frac{Y^T \sigma}{\varepsilon} & \text{sí } \|Y^T \sigma\| \leq \varepsilon \end{cases} \quad (5.23)$$

Donde $\varepsilon > 0$ y ρ es la cota de incertidumbre paramétrica, la cual es conocida *a priori*. La demostración detallada se puede observar en (Spong 1992).

En la figura 5.1 se presenta el esquema de la ley de control μ .

5.2 Ley de control *backstepping* con incertidumbre

El problema del control de sistemas lineales y no lineales con incertidumbre paramétrica es un aspecto ampliamente investigado y que ha generado diversos enfoques como: esquemas \mathcal{H}_∞ , esquemas robusto adaptivos, etc. (Sage *et al*, 1999; Casavola y Mosca, 1992; Li y Van Den Bosch, 1993). Otro enfoque para afrontar este

$$\mathbf{C} = \mathbf{C}_0 + \Delta \mathbf{C} \quad (5.26)$$

Sustituyendo (5.25) y (5.26) en (5.24) se obtiene:

$$\mathbf{M}_0(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{C}_0(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} + \mathbf{G} = \boldsymbol{\tau} + \Delta \quad (5.27)$$

Donde Δ representa la incertidumbre y las dinámicas no modeladas del robot.

A diferencia de Lotfesar y Chun-Yi (Lotfesar, *et al*, 2007; Lotfesar., *et al*, 2003; Chun-Yi y Yury, 1997; Chun-Yi, *et al*, 1996), los cuales presentan un diseño de un control *backstepping* el cual, para lograr realizar la representación del modelo del robot de forma como lo indican las ecuaciones (4.15) y (4.16), realizan una combinación del modelo dinámico del robot con el modelo de los motores del mismo; uno de los aportes de este trabajo es lograr obtener la representación del modelo del robot manipulador para que se pueda desarrollar un control *backstepping* sin la necesidad de tener en cuenta el modelo dinámico de los motores del robot, esto obedece a que en el proceso de identificación del un robot se tiene como resultado un modelo del manipulador que incluye los actuadores (motores)(Chan y Chen, 1999; Logothetis y Keiffer, 1996; Jagadeesh y Kurien, 2001) así que es necesario crear una forma de representar el modelo dinámico del robot para lograr aplicar un control *backstepping*, por otra parte una limitantes la de no contar con el robot para obtener los parámetros dinámicos del robot y de sus actuadores, procedimiento que no es muy simple.

El diseño de la estrategia de control *backstepping* con incertidumbre paramétrica parte de la definición de las variables que definirán la señal de control virtual del sistema, dicha señal de control virtual será el vector de error de posición.

Las ecuaciones (5.28) y (5.29) presentan respectivamente los vectores de error de posición y velocidad de las articulaciones del robot.

$$\mathbf{x}_1 = \mathbf{q} - \mathbf{q}_d \quad (5.28)$$

$$\dot{\mathbf{x}}_1 = \dot{\mathbf{q}} - \dot{\mathbf{q}}_d \quad (5.29)$$

Siguiendo la metodología de Spong, Zhu y Burkan (Spong, 1992; y Zhu, *et al.*, 1997; Burkan y Uzman 2005) se define el vector de corrección de velocidad como:

$$\mathbf{x}_2 = \dot{\mathbf{x}}_1 - \sigma = \dot{\mathbf{x}}_1 + \mathbf{K}\mathbf{x}_1 \quad (5.30)$$

donde \mathbf{K} es una matriz diagonal y definida positiva.

Realizando un sencillo proceso aritmético sobre (5.30) se obtiene:

$$\dot{\mathbf{x}}_1 = \mathbf{x}_2 - \mathbf{K}\mathbf{x}_1 \quad (5.31)$$

Derivando $\dot{\mathbf{x}}_2$ se obtiene:

$$\dot{\mathbf{x}}_2 = \dot{\mathbf{x}}_1 + \mathbf{K}\dot{\mathbf{x}}_1 = \dot{\boldsymbol{\phi}} - \dot{\boldsymbol{\phi}}_d + \mathbf{K}\dot{\mathbf{x}}_1 \quad (5.32)$$

Despejando $\dot{\boldsymbol{\phi}}$ de (5.27) y sustituyendo en (5.32) se obtiene:

$$\dot{\mathbf{x}}_2 = \mathbf{M}_0^{-1}(\boldsymbol{\tau} + \Delta - \mathbf{C}_0\dot{\boldsymbol{\phi}} - \mathbf{G}) - \dot{\boldsymbol{\phi}}_d + \mathbf{K}\dot{\mathbf{x}}_2 - \mathbf{K}^2\mathbf{x}_1 \quad (5.33)$$

Las ecuaciones (5.34) y (5.35) muestran la nueva representación del modelo dinámico del robot SCARA teniendo en cuenta las incertidumbres.

$$\dot{\mathbf{x}}_2 = \mathbf{M}_0^{-1}(\boldsymbol{\tau} + \Delta - \mathbf{C}_0\dot{\boldsymbol{\phi}} - \mathbf{G}) - \dot{\boldsymbol{\phi}}_d + \mathbf{K}\dot{\mathbf{x}}_2 - \mathbf{K}^2\mathbf{x}_1 \quad (5.34)$$

$$\dot{\mathbf{x}}_1 = -\mathbf{K}\mathbf{x}_1 + \mathbf{x}_2 \quad (5.35)$$

Dadas las ecuaciones (5.34) y (5.35) se puede observar claramente que se logra obtener un modelo el cual tiene como entrada virtual de control a \mathbf{x}_1 y cuya integral representa la señal de control real del sistema, tal como se debe presentar un sistema para diseñar el control *backstepping* con incertidumbre paramétrica.

Para obtener la ley de control que establezca el sistema se asume en primera instancia la función de Lyapunov V que cumple con (4.17), en otras palabras dicha función de Lyapunov demuestra la estabilidad del sistema (5.34) como se indica en el aparte 4.3.1.

Teniendo en cuenta el lema 4.2, se define la función candidata de Lyapunov $V_a(\mathbf{x})$ dada por:

$$V_a = V + \frac{1}{2}\mathbf{x}_1^T\mathbf{K}\mathbf{x}_1$$

Donde la función de Lyapunov candidata $V_a(\mathbf{x})$ que demuestra la estabilidad del sistema representado por (5.34) y (5.35) es:

$$V_a = \frac{1}{2}\mathbf{x}_2^T\mathbf{M}_0\mathbf{x}_2 + \frac{1}{2}\mathbf{x}_1^T\mathbf{K}\mathbf{x}_1 \quad (5.36)$$

Derivando la función de Lyapunov y sustituyendo (5.34) y (5.35) se tiene:

$$\begin{aligned}
\dot{V}_a^{\&} &= \mathbf{x}_2^T \mathbf{M}_0 \dot{\mathbf{x}}_2 + \frac{1}{2} \mathbf{x}_2^T \dot{\mathbf{M}}_0 \mathbf{x}_2 + \mathbf{x}_1^T \mathbf{K} \dot{\mathbf{x}}_1 \\
&= \mathbf{x}_2^T (\tau + \Delta - \mathbf{C}_0 \dot{\mathbf{q}}_d - \mathbf{G} - \mathbf{M}_0 \ddot{\mathbf{q}}_d + \mathbf{M}_0 \mathbf{K} \mathbf{x}_2 - \mathbf{M}_0 \mathbf{K}^2 \mathbf{x}_1) \\
&\quad + \frac{1}{2} \mathbf{x}_2^T \dot{\mathbf{M}}_0 \mathbf{x}_2 + \mathbf{x}_1^T \mathbf{K} (\mathbf{x}_2 - \mathbf{K} \mathbf{x}_1) \\
&= \mathbf{x}_2^T (\tau + \Delta - \mathbf{C}_0 (\mathbf{x}_2 + \dot{\mathbf{q}}_d - \mathbf{K} \mathbf{x}_1) - \mathbf{G} - \mathbf{M}_0 \ddot{\mathbf{q}}_d + \mathbf{M}_0 \mathbf{K} \mathbf{x}_2 - \mathbf{M}_0 \mathbf{K}^2 \mathbf{x}_1) \\
&\quad + \frac{1}{2} \mathbf{x}_2^T \dot{\mathbf{M}}_0 \mathbf{x}_2 + \mathbf{x}_1^T \mathbf{K} (\mathbf{x}_2 - \mathbf{K} \mathbf{x}_1) \\
&= -\mathbf{x}_2^T \mathbf{C}_0 \mathbf{x}_2 + \frac{1}{2} \mathbf{x}_2^T \dot{\mathbf{M}}_0 \mathbf{x}_2 + \mathbf{x}_2^T (\tau + \Delta - \mathbf{C}_0 (\dot{\mathbf{q}}_d - \mathbf{K} \mathbf{x}_1) - \mathbf{G} - \mathbf{M}_0 \ddot{\mathbf{q}}_d + \mathbf{M}_0 \mathbf{K} \mathbf{x}_2 - \mathbf{M}_0 \mathbf{K}^2 \mathbf{x}_1) \\
&\quad + \mathbf{x}_1^T \mathbf{K} (\mathbf{x}_2 - \mathbf{K} \mathbf{x}_1) \\
&= \frac{1}{2} \mathbf{x}_2^T (\dot{\mathbf{M}}_0 - 2\mathbf{C}_0) \mathbf{x}_2 + \mathbf{x}_2^T (\tau + \Delta - \mathbf{C}_0 (\dot{\mathbf{q}}_d - \mathbf{K} \mathbf{x}_1) - \mathbf{G} - \mathbf{M}_0 \ddot{\mathbf{q}}_d + \mathbf{M}_0 \mathbf{K} \mathbf{x}_2 - \mathbf{M}_0 \mathbf{K}^2 \mathbf{x}_1) \\
&\quad + \mathbf{x}_1^T \mathbf{K} (\mathbf{x}_2 - \mathbf{K} \mathbf{x}_1)
\end{aligned}$$

Aplicando la propiedad 5.2 se tiene:

$$\begin{aligned}
\dot{V}_a^{\&} &= \mathbf{x}_2^T (\tau + \Delta - \mathbf{C}_0 (\dot{\mathbf{q}}_d - \mathbf{K} \mathbf{x}_1) - \mathbf{G} - \mathbf{M}_0 \ddot{\mathbf{q}}_d + \mathbf{M}_0 \mathbf{K} \mathbf{x}_2 - \mathbf{M}_0 \mathbf{K}^2 \mathbf{x}_1) \\
&\quad + \mathbf{x}_1^T \mathbf{K} (\mathbf{x}_2 - \mathbf{K} \mathbf{x}_1)
\end{aligned} \tag{5.37}$$

Ahora se define una ley de control (τ) que garantice que $\dot{V}_a^{\&}$ sea definida negativa y así cumplir con el corolario 4.1, el cual garantiza la estabilidad asintótica global del sistema. La ecuación (5.38) presenta la ley de control.

$$\tau = \mathbf{M}_0 \ddot{\mathbf{q}}_d + \mathbf{C}_0 \dot{\mathbf{q}}_d - \|\Delta\|_{\infty} - \mathbf{K} \mathbf{x}_1 - \mathbf{M}_0 \mathbf{K} \mathbf{x}_2 - \mathbf{G} \tag{5.38}$$

Sustituyendo (5.38) en (5.37) se obtiene

$$\begin{aligned}
\dot{V}_a^{\&} &= \mathbf{x}_2^T (-\mathbf{C}_0 \mathbf{K} \mathbf{x}_1 - \mathbf{M}_0 \mathbf{K}^2 \mathbf{x}_1 - \mathbf{K} \mathbf{x}_1 - \|\Delta\|_{\infty} + \Delta) \\
&\quad + \mathbf{x}_1^T \mathbf{K} (\mathbf{x}_2 - \mathbf{K} \mathbf{x}_1) \\
&= -\mathbf{x}_2^T \mathbf{C}_0 \mathbf{K} \mathbf{x}_1 - \mathbf{x}_2^T \mathbf{M}_0 \mathbf{K}^2 \mathbf{x}_1 - \mathbf{x}_2^T \mathbf{K} \mathbf{x}_1 - \mathbf{x}_2^T \|\Delta\|_{\infty} + \mathbf{x}_2^T \Delta \\
&\quad + \mathbf{x}_1^T \mathbf{K} \mathbf{x}_2 - \mathbf{x}_1^T \mathbf{K}^2 \mathbf{x}_1
\end{aligned} \tag{5.39}$$

Aplicando norma a unos sumandos de (5.39) y aplicando la propiedad 5.3 se tiene:

$$\begin{aligned}
 \dot{V}_a &= -\mathbf{x}_2^T \mathbf{C}_0 \mathbf{K} \mathbf{x}_1 - \mathbf{x}_2^T \mathbf{M}_0 \mathbf{K}^2 \mathbf{x}_1 - \|\mathbf{x}_2^T \mathbf{K} \mathbf{x}_1\|_\infty \\
 &\quad - \|\mathbf{x}_2^T \|\Delta\|_\infty\|_\infty + \|\mathbf{x}_2^T \Delta\|_\infty + \|\mathbf{x}_1^T \mathbf{K} \mathbf{x}_2\|_\infty - \mathbf{x}_1^T \mathbf{K}^2 \mathbf{x}_1 \\
 &\leq -\mathbf{x}_2^T \mathbf{C}_0 \mathbf{K} \mathbf{x}_1 - \mathbf{x}_2^T \mathbf{M}_0 \mathbf{K}^2 \mathbf{x}_1 - \|\mathbf{x}_2^T\|_\infty \|\mathbf{K} \mathbf{x}_1\|_\infty \\
 &\quad - \|\mathbf{x}_2^T\|_\infty \|\Delta\|_\infty + \|\mathbf{x}_2^T\|_\infty \|\Delta\|_\infty + \|\mathbf{x}_1^T\|_\infty \|\mathbf{K} \mathbf{x}_2\|_\infty - \mathbf{x}_1^T \mathbf{K}^2 \mathbf{x}_1 \\
 &\leq -\mathbf{x}_2^T \mathbf{C}_0 \mathbf{K} \mathbf{x}_1 - \mathbf{x}_2^T \mathbf{M}_0 \mathbf{K}^2 \mathbf{x}_1 - \mathbf{x}_1^T \mathbf{K}^2 \mathbf{x}_1 \\
 &\leq -(\mathbf{C}_0 \mathbf{K} + \mathbf{M}_0 \mathbf{K}^2) \|\mathbf{x}_2^T\| \|\mathbf{x}_1\| - \mathbf{K}^2 \|\mathbf{x}_1\|^2 < 0
 \end{aligned}
 \tag{5.40}$$

Se puede observar en (5.40) que \dot{V}_a es definida negativa, luego la ley de control definida en (5.38) hace que el sistema sea globalmente asintóticamente estable. Es importante resaltar que la ley de control obtenida para el robot SCARA es simple de implementar como se puede observar en la figura 5.2, y la cual depende del modelo nominal del robot.

En la figura 5.2 se observa el esquema del control *backstepping* con incertidumbre para el robot SCARA de cuatro grados de libertad.

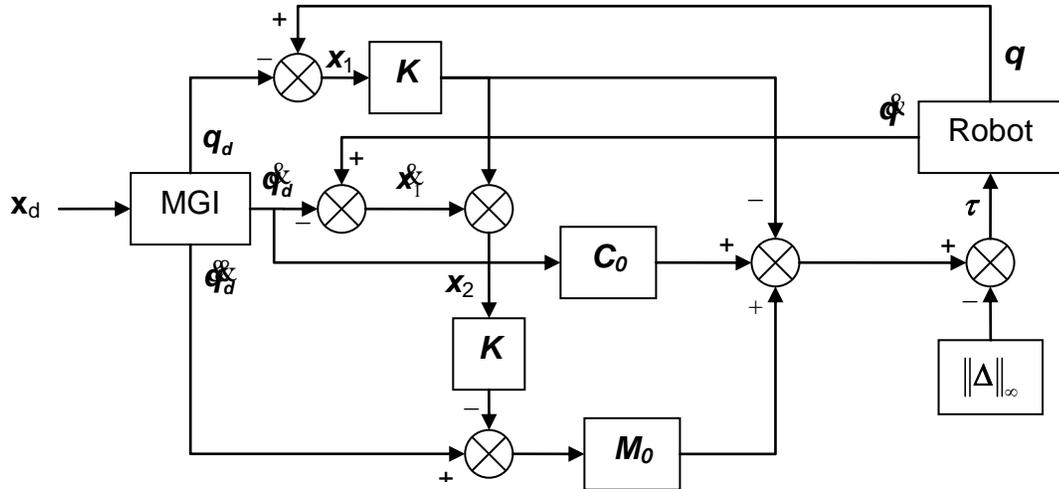


Figura 5.2: Esquema control *backstepping*.

En el esquema de la figura 5.2, X_d representa las trayectorias en x , y y z que el efector final del robot SCARA debe seguir.

Una diferencia entre las estrategias de control desarrolladas en este capítulo es la simplicidad del control *backstepping* frente a la técnica propuesta por Spong, por otra parte el control *backstepping* requiere de 4 parámetros para la sintonización (matriz \mathbf{K}) mientras para la técnica de Spong es necesario determinar 8 parámetros (matriz \mathbf{K} y Δ), lo cual presenta una mayor dificultad en el proceso de sintonización.

Capítulo 6

Resultados de simulación

En este capítulo se presentan los resultados a nivel de simulación de las estrategias de control por par calculado (*CTC*), control robusto de Spong (Spong, 1992) y control *Backstepping* con incertidumbre, las cuales fueron desarrolladas en capítulos anteriores. Las pruebas realizadas para las estrategias de control se basan en el seguimiento de las trayectorias circular y lineal que se presentaron en el capítulo 3; además se realizan simulaciones teniendo en cuenta la incertidumbre paramétrica y el aumento en la velocidad del robot al desarrollar el trabajo definido.

Los resultados presentados son: el error de seguimiento para la trayectoria circular en estado estacionario y estado estable, error de seguimiento para cambios de dirección, para lo cual se utiliza la trayectoria lineal y finalmente se presenta los esfuerzos de control.

6.1 Esquema en *Simulink* de las estrategias de control

6.1.1 Esquema de la ley de control robusta de Spong

En la figura 6.1 se presenta el esquema en *Simulink* de la ley de control representada en la ecuación (5.23), en este esquema se pueden observar los siguientes bloques:

- *Function Y*: Ejecuta la función llamada *Y_4gl.m* de *Matlab* la cual calcula la matriz Y que se presenta en la ecuación (5.6).
- *Function Tau*: Calcula la ley de control expresada en la ecuación (5.23).
- *PI0*: Es el vector de parámetros π_0 .
- *Delta*: Representa la matriz Δ .

El esquema en *Simulink* del robot SCARA controlado por la ley de control robusta de Spong es igual al presentado en la figura 3.3, la diferencia está en que el bloque de control PID se cambia por el esquema presentado en la figura 6.1.

Las funciones y diagramas que se desarrollan para la simulación de esta estrategia de control se encuentran en el anexo A.

El proceso de sintonización del controlador se realiza por medio de la selección de los valores de las matrices diagonales y definidas positivas \mathbf{K} y Λ , cuyos valores se muestran en la tabla 6.1.

Tabla 6.1: Coeficientes de las matrices \mathbf{K} y Λ

	Articulación 1	Articulación 2	Articulación 3	Articulación 4
\mathbf{K}	1200	200	200	1500
Λ	1550	550	70	1500

Estas matrices se sintonizan por medio del bloque *Signal Constrain* de *Simulink* y posteriormente se realiza una sintonización fina para lograr mejores resultados en lo referente al error de seguimiento.

6.1.2 Esquema de la ley de control *backstepping* con incertidumbre.

El esquema en *Simulink* del control *backstepping* con incertidumbre se presenta en la figura 6.2 en la cual se pueden identificar:

- *Fuction Norma*: Realiza el cálculo de la norma Frobenius de la Matriz \mathbf{M} .
- *Fuction M*: Ejecuta la función llamada Matriz_M_4gl.m, la cual realiza los cálculos en línea de la matriz \mathbf{M} con lo valores nominales de los parámetros inerciales.

El esquema en *Simulink* del robot SCARA controlado por la ley de control *backstepping* con incertidumbre es similar al presentado en la figura 3.3, la diferencia está en que el bloque de control PID se sustituye por el esquema presentado en la figura 6.2.

Las funciones y diagramas que se desarrollan para la simulación de esta estrategia de control se encuentran en el anexo A.

El proceso de sintonización del controlador se realiza por medio de la selección de los valores de la matriz diagonal y definida positiva \mathbf{K} , dicha sintonización se realizó por medio del bloque *Signal Constrain* de *Simulink* y posteriormente se efectúa una sintonización fina para lograr mejores resultados en lo referente al error de seguimiento, los valores de la matriz \mathbf{K} son de 800.

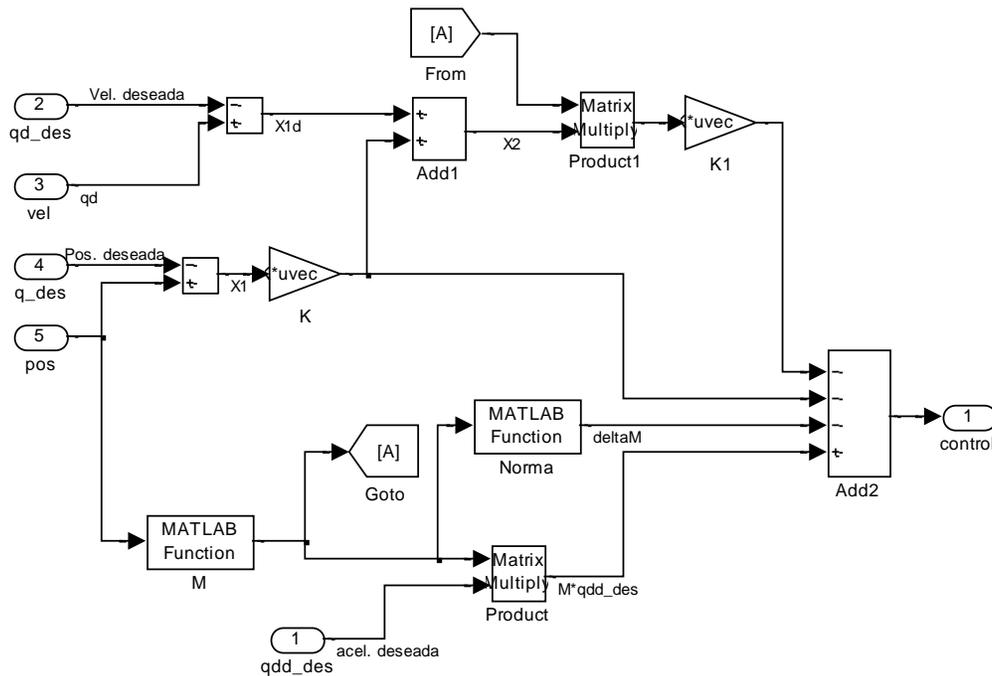


Figura 6.2: Esquema en Simulink del control *backstepping* con incertidumbre.

6.2 Resultados de simulación.

En este aparte se presentan los resultados de simulación de las estrategias Control CTC, control Robusto de Spong (CRS) y Control *Backstepping* con Incertidumbre (CBI). Es importante resaltar que el proceso de sintonización de los controladores se basó en el error de seguimiento a las trayectorias descritas en el capítulo 3 (trayectoria circular y lineal) las cuales se ejecutan en 5 y 3 segundos; la incertidumbre paramétrica se tiene en cuenta en el proceso de simulación.

6.2.1 Resultados de simulación con incertidumbre

Las figuras 6.3 y 6.4 muestran el error de seguimiento a la trayectoria circular ejecutada en 5 segundos en estado transitorio y estacionario de las estrategias de control robusto de Spong y control *backstepping* con incertidumbre, para esta simulación no se tuvo en cuenta la incertidumbre, en otras palabras se aplicó las estrategias de control al sistema nominal.

Como se puede observar, el desempeño del controlador *backstepping* sin incertidumbre presenta una mejor respuesta en cuanto al error de seguimiento,

presentando un error de 5.5×10^{-5} m en estado transitorio y 3.5×10^{-7} m en estado estacionario.

La figura 6.5 presenta el error de seguimiento para el cambio de dirección en la trayectoria lineal para el control CRS y CBI. Los resultados de esta simulación muestran que el controlador CBI tiene mejor desempeño presentando un error de seguimiento de 2×10^{-5} metros frente a 5.5×10^{-4} del control CRS.

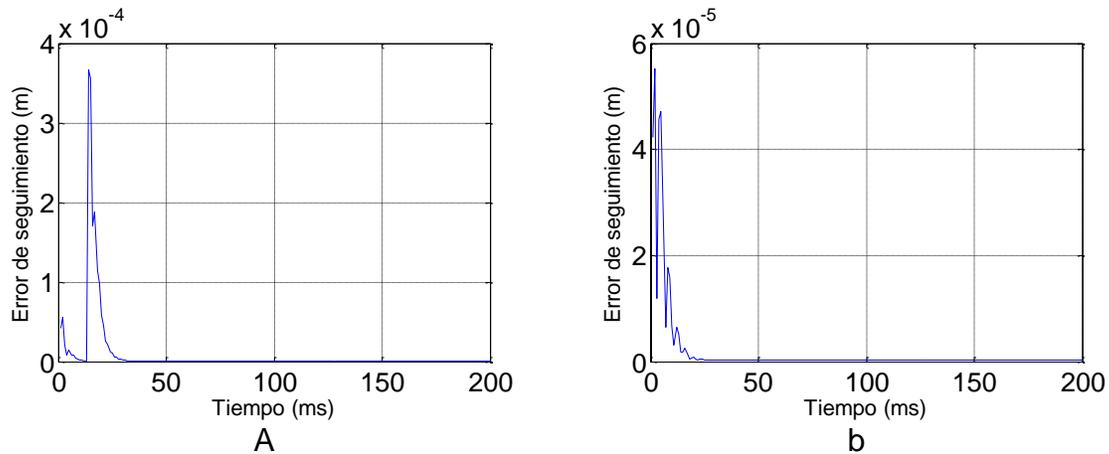


Figura 6.3: Error de seguimiento, trayectoria circular 5 seg. (transitorio).
a) CRS, b) CBI.

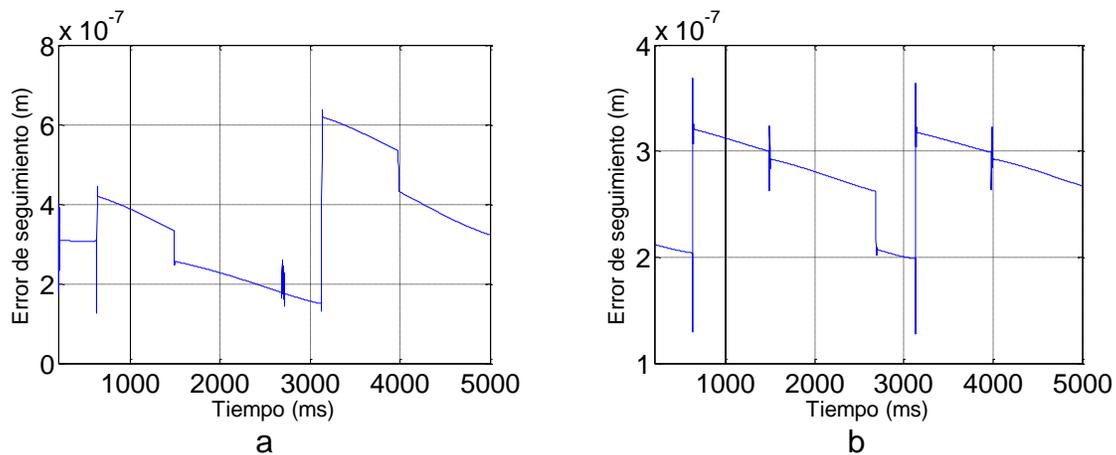


Figura 6.4: Error de seguimiento, trayectoria circular 5 seg. (estacionario).
a) CRS, b) CBI.

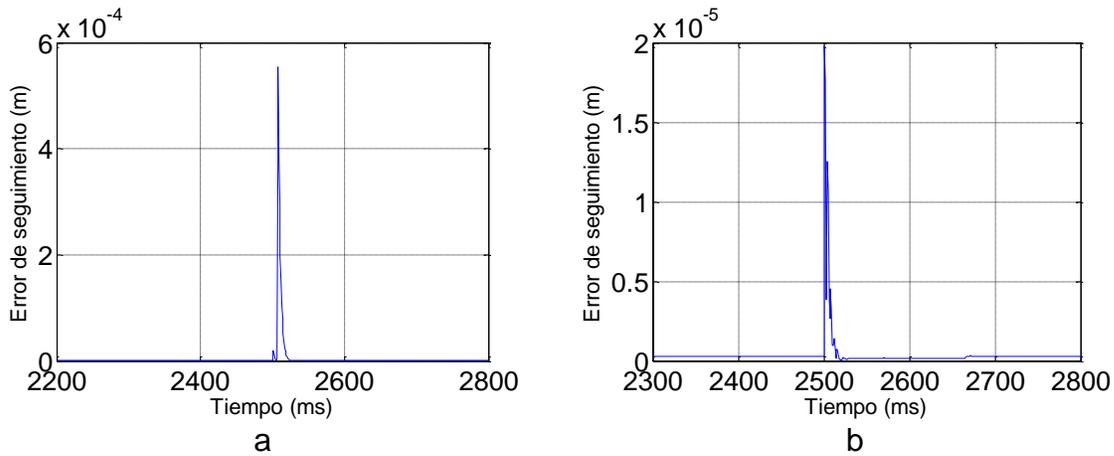


Figura 6.5: Error de seguimiento, trayectoria lineal 5 seg. (cambio de dirección). a) CRS, b) CBI.

A continuación se presentan los resultados de simulación de las dos estrategias de control robustas, teniendo en cuenta la incertidumbre paramétrica; para la simulación del robot con incertidumbre se afecta los parámetros dinámicos del robot SCARA por un factor constante para todos, por ejemplo, se varía el vector de parámetros en un 10% ó 20%.

Las figuras 6.6, 6.7 y 6.8 muestran los resultados de simulación obtenidos para una incertidumbre de un 10% en los parámetros del robot SCARA, para las trayectorias circular y lineal ejecutadas en 5 segundos.

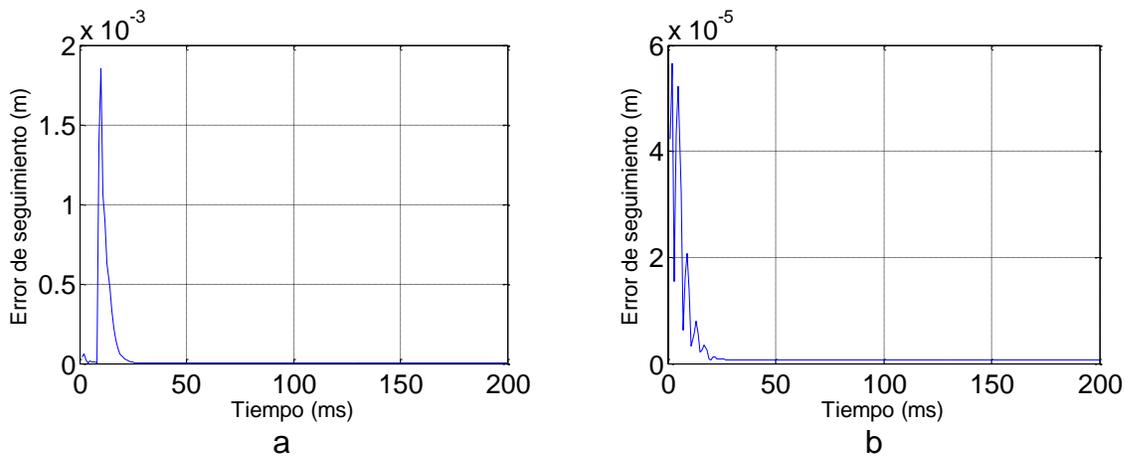


Figura 6.6: Error de seguimiento, trayectoria circular 5 seg. (transitorio) con incertidumbre de 10%. a) CRS, b) CBI.

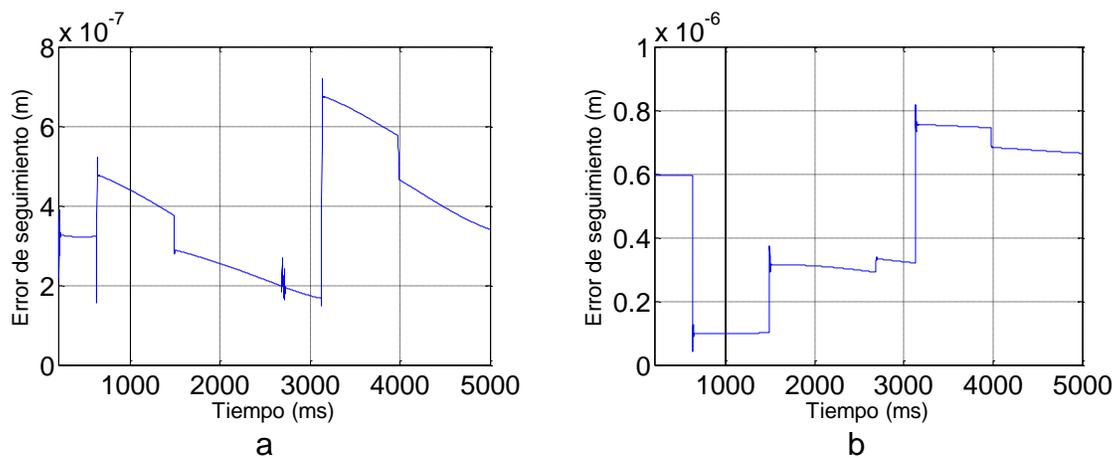


Figura 6.7: Error de seguimiento, trayectoria circular 5 seg. (estacionario) con incertidumbre de 10% a) CRS, b) CBI.

En el error de seguimiento para la trayectoria circular se puede observar que el controlador CRS presenta un mejor desempeño que la estrategia CBI, con respecto al estado estacionario, presentando un error de 7.2×10^{-7} m frente a 8.1×10^{-7} m del control CBI, en estado transitorio del control CBI presenta un error de 5.6×10^{-5} frente a 1.8×10^{-3} m del CRS. En lo referente al cambio de dirección ambas estrategias de control presentan un error muy similar frente al caso sin incertidumbre, igualmente la respuesta es claramente más suave para el control CRS.

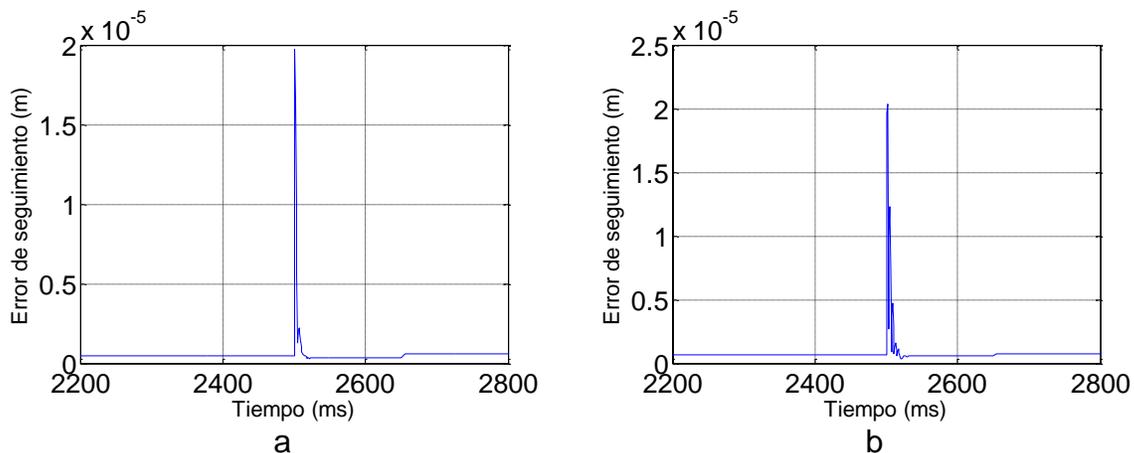


Figura 6.8: Error de seguimiento, trayectoria lineal 5 seg. (cambio de dirección) con incertidumbre de 10% a) CRBL, b) CBI.

Las figuras 6.9, 6.10 y 6.11 presentan los resultados de simulación del control CRS y el control CBI, para una incertidumbre del 20%. Se observa en estas figuras que el control CBI presenta mejor desempeño que el control CRS en estado transitorio con un error de seguimiento de 5.7×10^{-5} frente a 3.2×10^{-4} del control CRS, en estado

estacionario el control CBI presenta un error de 1.3×10^{-6} frente a un error de 0.8×10^{-6} m del control CRS.

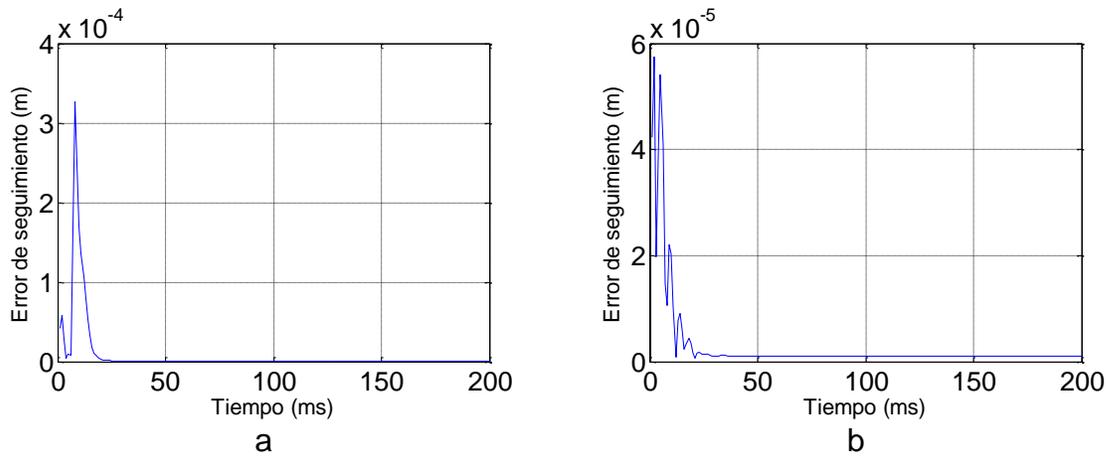


Figura 6.9: Error de seguimiento, trayectoria circular 5 seg. (transitorio) con incertidumbre de 20% a) CRS, b) CBI.

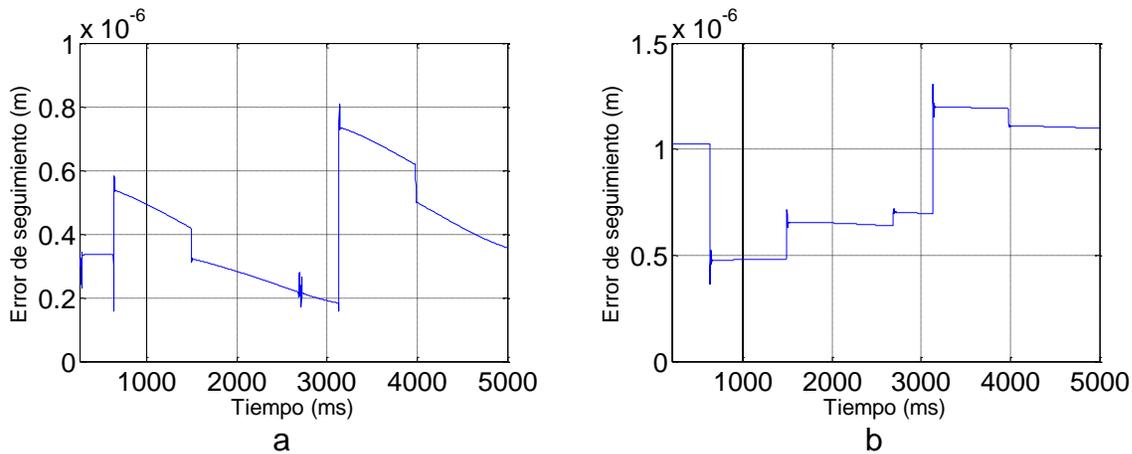


Figura 6.10: Error de seguimiento, trayectoria circular 5 seg. (estacionario) con incertidumbre de 20% a) CRS, b) CBI.

Para el cambio brusco de trayectoria, el control CRS presenta un error de 2×10^{-5} mientras que el control CBI tiene un error de 2.2×10^{-5} .

La tabla 6.2 presenta un resumen de los resultados de simulación de las estrategias de control *CTC*, *CRS* y *CBI* frente a incertidumbres en los parámetros del modelo. Para estas pruebas no se tuvo en cuenta el control *PID* debido a que no es una estrategia por modelo de referencia, las estrategias de control *CTC*, *CRS* y *CBI* por el contrario dependen del modelo del robot, lo cual implica que la señal de control está ligada a las variaciones en los parámetros del manipulador (Khalil y Dombre, 2002; Spong 1992), luego el desempeño del control depende de la diferencia entre el

modelo real del manipulador y el obtenido por medio de los procesos expresados en el capítulo 2 y teniendo en cuenta los parámetros inerciales nominales

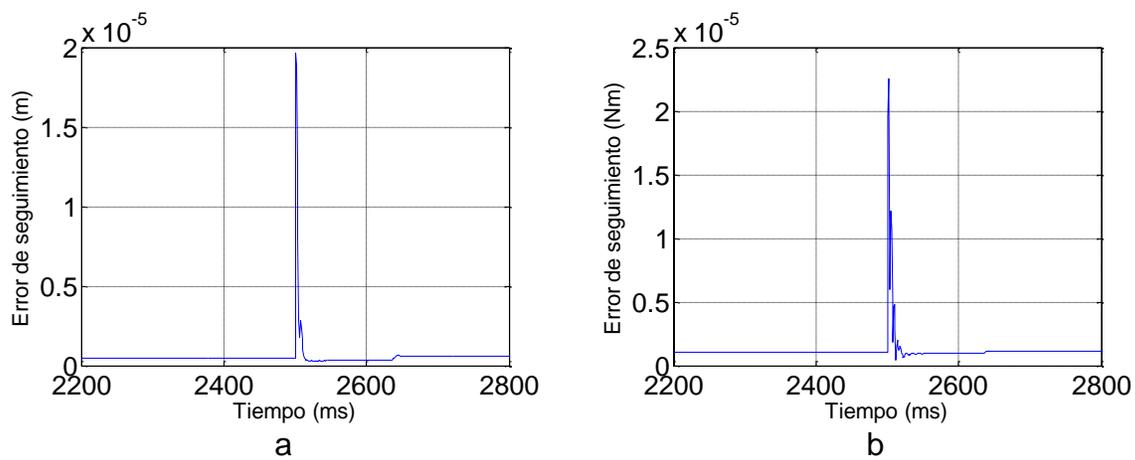


Figura 6.11: Error de seguimiento, trayectoria lineal 5 seg. (cambio de dirección) con incertidumbre de 20% a) CRS, b) CBI.

Tabla 6.2: Error de seguimiento, CTC, CRS y CBI, para un tiempo de ejecución de 5 segundos.

% de incertid.	Error de Seguimiento CTC (m)			Error de seguimiento CRS (m)			Error de seguimiento CBI (m)		
	Circ. Trans.	Circ. Esta.	Lineal	Circ. Trans.	Circ. Esta.	Lineal	Circ. Trans.	Circ. Esta.	Lineal
0	1.8×10^{-4}	1.3×10^{-6}	2.7×10^{-5}	3.65×10^{-4}	6.4×10^{-7}	5.5×10^{-4}	5.5×10^{-5}	3.7×10^{-7}	2×10^{-5}
10	3.4×10^{-4}	1×10^{-6}	2.8×10^{-5}	1.8×10^{-3}	7.2×10^{-7}	2×10^{-5}	5.6×10^{-5}	8.1×10^{-7}	2×10^{-5}
-10	3.1×10^{-4}	1.1×10^{-6}	2.5×10^{-5}	Inestable			5.4×10^{-5}	7.2×10^{-7}	2×10^{-5}
20	3.4×10^{-4}	9.5×10^{-7}	3×10^{-5}	1.7×10^{-4}	8×10^{-7}	2.1×10^{-5}	5.7×10^{-5}	1.3×10^{-6}	2.2×10^{-5}
-20	5.2×10^{-5}	1×10^{-6}	2.2×10^{-5}	Inestable			5.1×10^{-5}	1×10^{-6}	2×10^{-5}
40	1.7×10^{-4}	1×10^{-6}	3.1×10^{-5}	9.1×10^{-4}	1×10^{-6}	2.2×10^{-5}	2.7×10^{-5}	2.2×10^{-6}	2.5×10^{-5}
-40	Inestable			Inestable			Inestable		
60	1.7×10^{-4}	1.2×10^{-6}	3.6×10^{-5}	1.41×10^{-3}	1.2×10^{-6}	2.2×10^{-4}	7.8×10^{-5}	3.2×10^{-6}	2.7×10^{-5}
-60	Inestable			Inestable			Inestable		
100	1.7×10^{-4}	1.3×10^{-6}	4.3×10^{-5}	1.7×10^{-3}	1.8×10^{-6}	3×10^{-5}	2.1×10^{-4}	5.2×10^{-6}	3.2×10^{-5}

De la tabla se puede observar que el control CRS se inestabiliza para incertidumbres negativas, mientras que el control CBI soporta un mayor rango negativo en la incertidumbre al igual que el control CTC. Con respecto a los resultados con porcentaje positivo en la variación de la incertidumbre, es claro que el desempeño del CBI es superior en lo referente al error de seguimiento en estado transitorio de la trayectoria circular, mientras que en estado estacionario para una incertidumbre del 40 y 100%, el control CTC presenta mejor desempeño. Por último en lo referente al

cambio de dirección, el desempeño del control CBI es mejor que el control *CTC*; con respecto al control CRS, éste presenta mejor desempeño que el CBI para incertidumbres superiores al 20% aunque no es muy grande esta diferencia; se puede observar también que el control CRS presenta una respuesta más suave que el control CBI, lo cual indica un mejor desempeño del control CRS frente a cambios de dirección.

Estos resultados presentados anteriormente muestran una superioridad del control CBI frente al error de seguimiento en estado transitorio. Esto se debe a que el control CRS es una técnica que estabiliza un sistema no lineal fuertemente acoplado lo cual hace más 'difícil' la tarea para el controlador, pero la técnica *backstepping*, que está estrechamente relacionada con la linealización por realimentación con la diferencia que la metodología del *backstepping* no elimina todas las no linealidades sino que elimina únicamente las desestabilizantes, permite que el controlador tenga una tarea más 'fácil' para hacer el control de seguimiento.

La suavidad de la respuesta en el cambio de dirección por parte del control CRS se debe a la regulación que se realiza en los vectores de velocidad y aceleración, lo cual reprovee a esta estrategia una mejor reacción frente a cambios bruscos de dirección; esta regulación no se tuvo en cuenta en el control CBI con el fin de obtener una ley de control computacionalmente más 'liviana', pensando en un tiempo de ejecución más corto en el momento de la implementación del controlador.

6.2.2 Resultados de simulación para mayor velocidad de ejecución

En esta sección se presenta el error de seguimiento producido por las estrategias de control *CTC*, CRS y CBI para las trayectorias circular y lineal descritas anteriormente, pero para un tiempo de ejecución menor, en este caso 3 segundos. El objetivo de estas pruebas es determinar el desempeño de las técnicas de control frente al aumento de velocidad en el desarrollo de una consigna cartesiana. Los cambios realizados en las trayectorias se pueden observar en el anexo A, en el cual se encuentra el programa en *Matlab* denominado Circular_4gl.m y Lineal_4gl.m.

La figuras 6.12, 6.13 y 6.14 presentan los resultados de simulación del sistema sin incertidumbre para las estrategias de control CRS y CBI. Y las figuras 6.15, 6.16 y 6.17 muestran el error de seguimiento de las estrategias de control CRS y CBI para una incertidumbre del 10%.

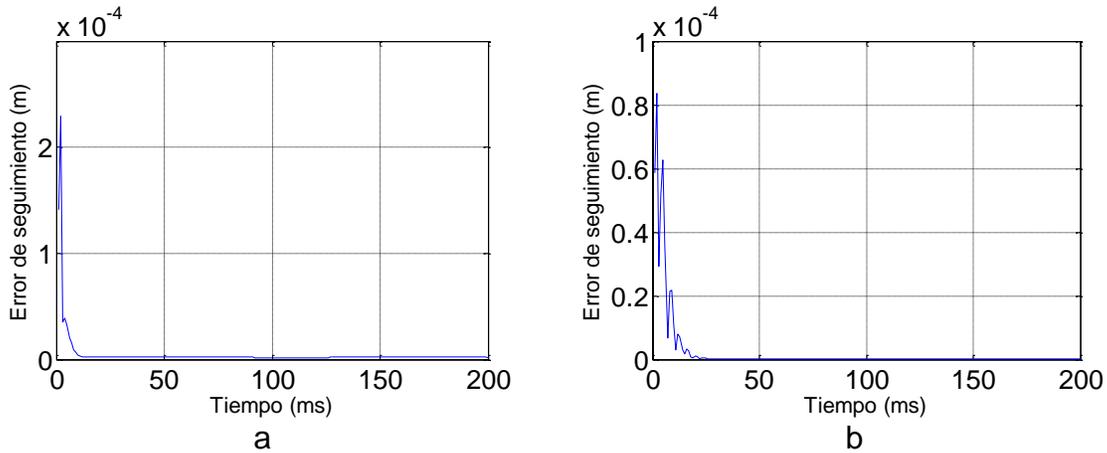


Figura 6.12: Error de seguimiento, trayectoria circular 3 seg. (transitorio) sin incertidumbre a) CRS, b) CBI.

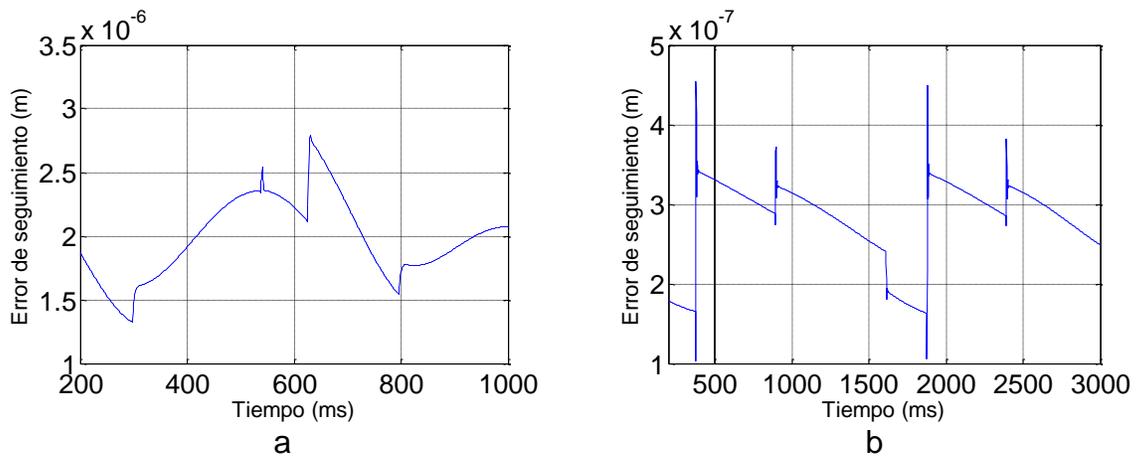


Figura 6.13: Error de seguimiento, trayectoria circular 3 seg. (estacionario) sin incertidumbre a) CRS, b) CBI.

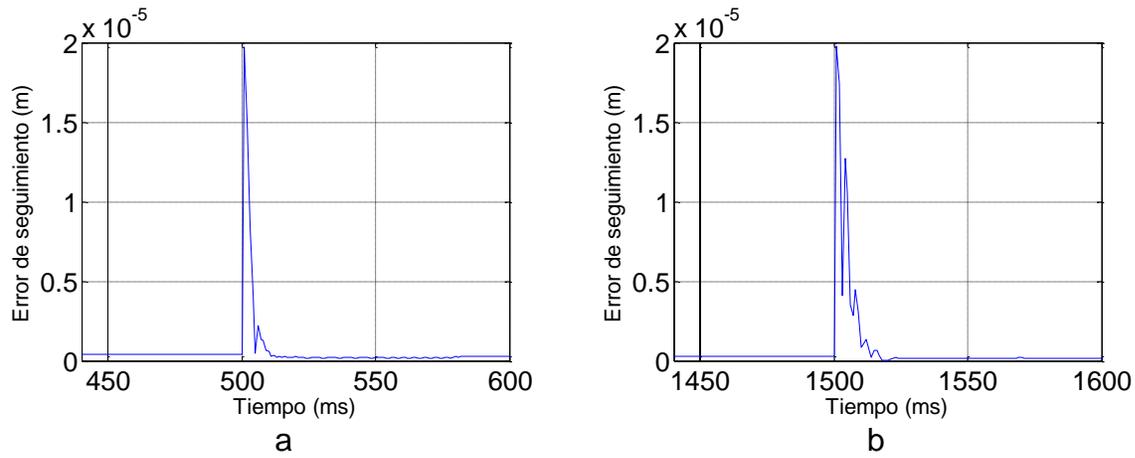


Figura 6.14: Error de seguimiento, trayectoria lineal 3 seg. (cambio de dirección) sin incertidumbre a) CRS, b) CBI.

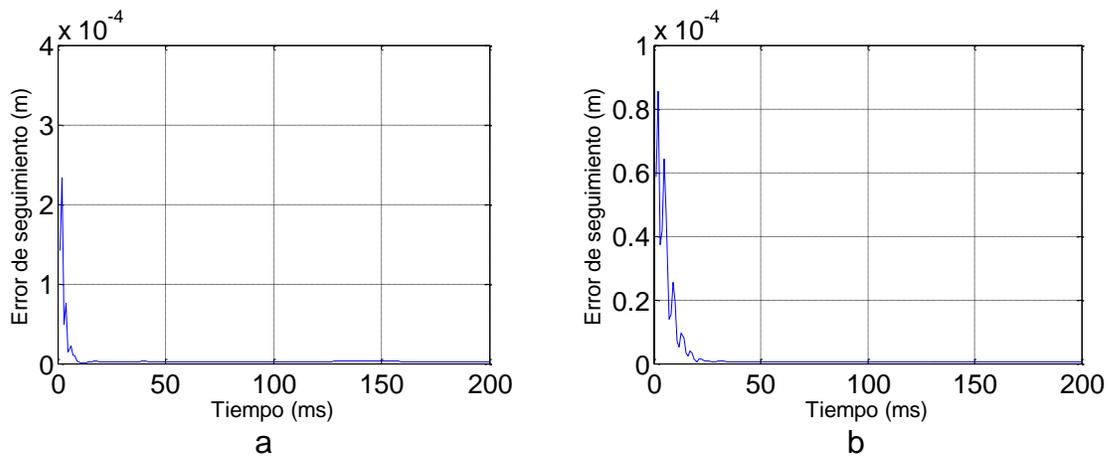


Figura 6.15: Error de seguimiento, trayectoria circular 3 seg. (transitorio) con incertidumbre de 10%. a) CRS, b) CBI.

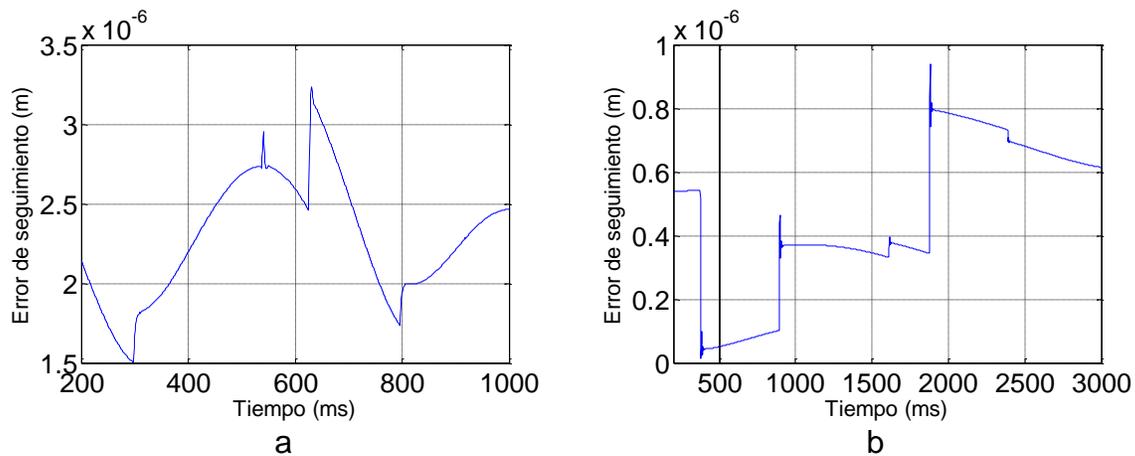


Figura 6.16: Error de seguimiento, trayectoria circular 3 seg. (estacionario) con incertidumbre de 10%. a) CRS, b) CBI.

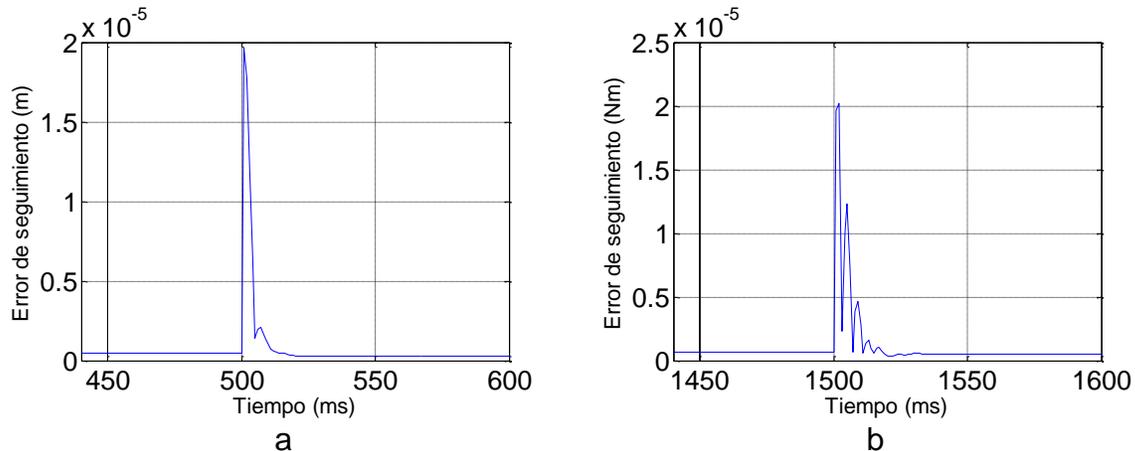


Figura 6.17: Error de seguimiento, trayectoria lineal 3 seg. (cambio de dirección) con incertidumbre de 10%. a) CRS, b) CBI.

De las figuras anteriores y comparando con las figuras 6.6, 6.7 y 6.8 se puede observar que el desempeño del control CRS mejora en relación al error en estado transitorio, para el control CBI su desempeño se incrementa pero sigue siendo mejor que el control CRS. Con respecto al cambio de dirección ambas estrategias de control presentan un desempeño similar para incertidumbre del 10 y 20%, mientras que para el sistema nominal el control CRS mejora notablemente en este aspecto pasando de 5.5×10^{-4} a 2×10^{-5} . El error de seguimiento en estado estacionario el control CBI presenta un mejor desempeño para incertidumbre del 10% y para el sistema nominal.

La tabla 6.3 presenta los resultados de simulación de las estrategias de control teniendo en cuenta la incertidumbre.

Los resultados consignados en la tabla 6.3 muestran un resultado similar a los presentados en la tabla 6.2 en cuanto a porcentajes de incertidumbre negativos. Para la trayectoria circular se puede observar que el control CBI es mejor en el error de seguimiento en estado transitorio para todos los porcentajes positivos de incertidumbre, mientras que para el estado estacionario el control CRS presenta mejor respuesta exceptuando el caso del sistema sin incertidumbre. En lo referente al cambio de dirección, el desempeño en ambas estrategias de control es similar entre ellas y a su vez similar al error presentado cuando el tiempo de ejecución fue de 5 segundos.

Con respecto al control *CTC*, éste presenta una mejor respuesta en estado transitorio para la trayectoria circular cuando la incertidumbre supera el 40%, en cuanto al error de estado estacionario y al error para el cambio de dirección, presenta menor desempeño que las otras estrategias de control.

Si se compara los resultados de las tablas 6.2 y 6.3 se puede observar que el control CBI y CRS mejoran en algunos aspectos, como en el error de seguimiento de la trayectoria circular, pero aunque se ven afectados por el aumento de la velocidad, el desempeño que presentan estas dos estrategias de control son similares a los resultados obtenidos anteriormente con la ejecución de las trayectorias de 5 segundos. Con respecto al control *CTC*, la ley de control CRS presenta mejor desempeño en el error de seguimiento en estado transitorio cuando se aumenta la velocidad de ejecución, para el error en estado estacionario aunque el control *CTC* tiene mejor desempeño la diferencia con el control CBI no es muy grande, con respecto al cambio de dirección el control CBI tiene un mejor desempeño frente al control *CTC* cuando se incrementa la velocidad.

Tabla 6.3: Error de seguimiento, *CTC*, CRS y CBI para tiempo de ejecución de 3 seg.

% de incertid.	Error de Seguimiento CTC (m)			Error de seguimiento CRS (m)			Error de seguimiento CBI (m)		
	Circ. Trans.	Circ. Esta.	Lineal	Circ. Trans.	Circ. Esta.	Lineal	Circ. Trans.	Circ. Esta.	Lineal
0	1.4×10^{-3}	2×10^{-6}	2.5×10^{-4}	2.5×10^{-4}	8×10^{-7}	2×10^{-5}	8.4×10^{-5}	3.61×10^{-7}	2×10^{-5}
10	1.2×10^{-3}	2×10^{-6}	2.6×10^{-4}	6.2×10^{-4}	9×10^{-6}	2×10^{-5}	8.5×10^{-5}	9.4×10^{-7}	2×10^{-5}
-10	5.1×10^{-3}	2×10^{-6}	2.2×10^{-4}	Inestable		2×10^{-5}	7.9×10^{-5}	6.8×10^{-7}	2×10^{-5}
20	4.2×10^{-3}	2×10^{-6}	2.6×10^{-4}	3.3×10^{-3}	1×10^{-6}	1.9×10^{-5}	8.7×10^{-5}	1.45×10^{-6}	2.2×10^{-5}
-20	5×10^{-3}	2×10^{-6}	2×10^{-4}	Inestable			7.9×10^{-5}	1×10^{-6}	2×10^{-5}
40	2.5×10^{-4}	2.1×10^{-6}	3×10^{-4}	16.5×10^{-4}	1.4×10^{-6}	2.8×10^{-5}	4.6×10^{-5}	2.4×10^{-6}	2.5×10^{-5}
-40	Inestable			Inestable			Inestable		
60	3.5×10^{-3}	2.3×10^{-6}	3×10^{-4}	1.05×10^{-3}	1.7×10^{-6}	2.6×10^{-5}	5.4×10^{-5}	3.5×10^{-6}	2.7×10^{-5}
-60	Inestable			Inestable			Inestable		
100	8×10^{-4}	3×10^{-6}	3.9×10^{-4}	1.7×10^{-3}	2.6×10^{-6}	3.1×10^{-5}	3.6×10^{-4}	5.5×10^{-6}	3.2×10^{-5}

6.2.3 Esfuerzo de control

Otro aspecto importante para analizar en el desempeño, es el esfuerzo de control. Para el caso del control del robot *SCARA*, se presenta en esta sección el par generado por los motores de cada articulación para lograr seguir las trayectorias definidas previamente en el espacio cartesiano (trayectoria circular y lineal) para un tiempo de ejecución de 5 y 3 segundos.

Es importante resaltar que en los resultados presentados a continuación no se tiene en cuenta la ganancia de los actuadores, debido a que no se posee esta información, la cual se obtiene por medio de un proceso de identificación con el robot real (Restrepo, 1996). Si se dispusiera de este valor, el esfuerzo de control debería ser dividido por esta ganancia identificada.

Los resultados presentados en las figuras 6.18 a 6.23 muestran el esfuerzo de control para la trayectoria circular, para el sistema sin incertidumbre y con una incertidumbre del 10 y del 20% para un tiempo de ejecución de 5 segundos.

Se puede observar claramente que el control CBI presenta un mejor desempeño en cuanto al esfuerzo de control en estado transitorio frente a control CRS. Es claro que en estado estacionario ambas estrategias de control presentan un esfuerzo muy similar. En la tabla 6.4 se presentan los valores pico de los esfuerzos de control del control *CTC*, *CRS* y *CBI*.

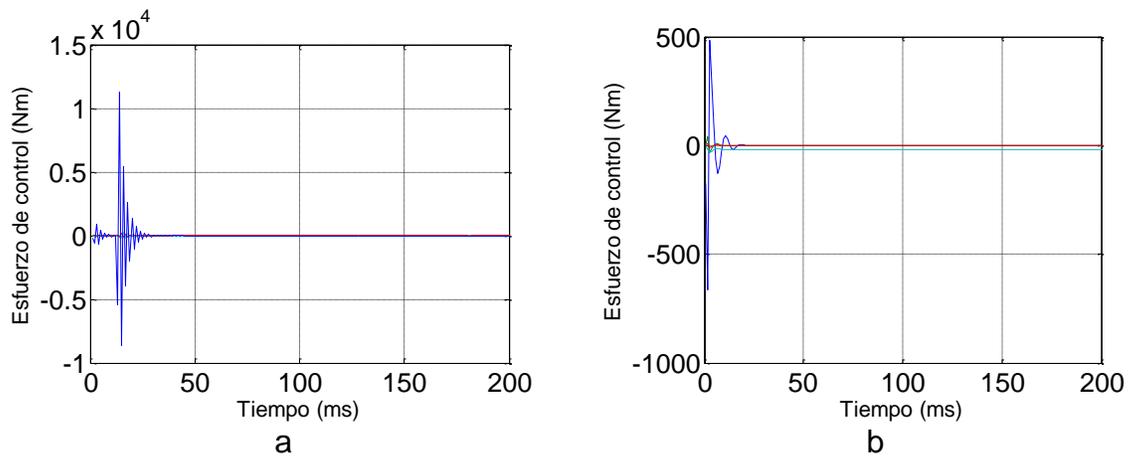


Figura 6.18: Par generado por los motores sin incertidumbre 5 seg. (transitorio). a) CRS, b) CBI.

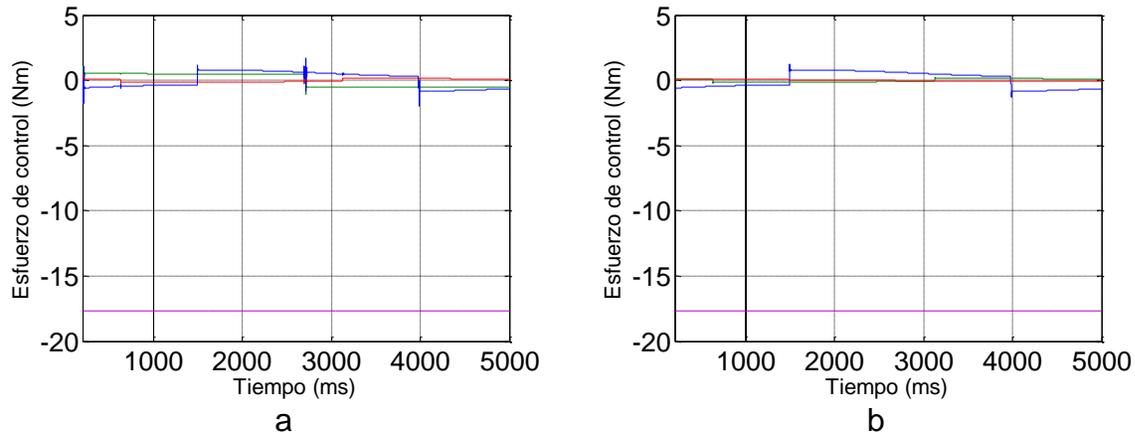


Figura 6.19: Par generado por los motores sin incertidumbre 5 seg. (estacionario) a) CRS, b) CBI.

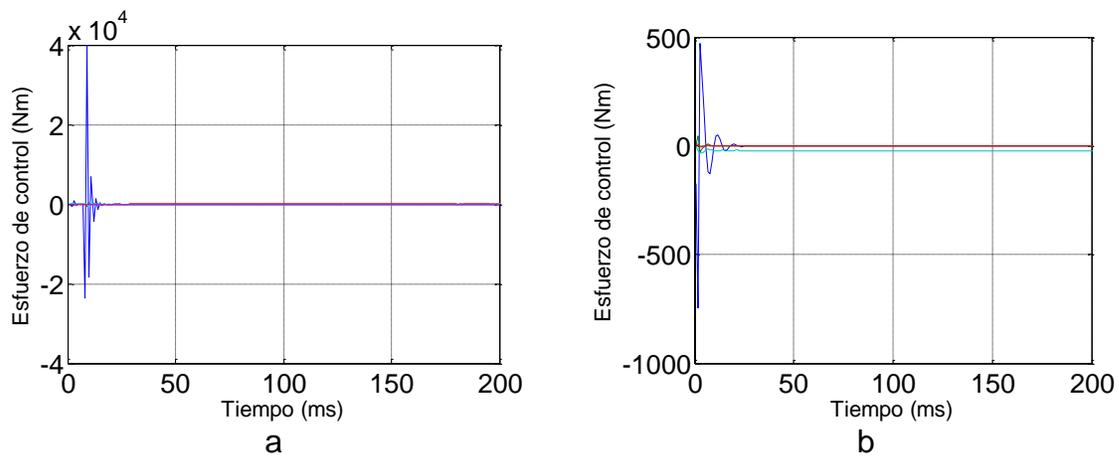


Figura 6.20: Par generado por los motores con incertidumbre del 10% 5 seg. (transitorio) a) CRS, b) CBI.

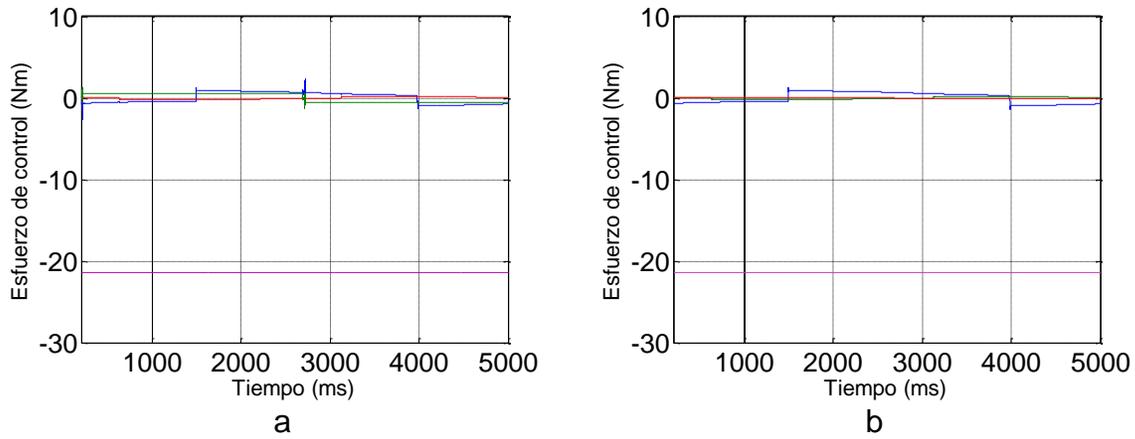


Figura 6.21: Par generado por los motores con incertidumbre del 10% 5 seg. (estacionario) a) CRS, b) CBI.

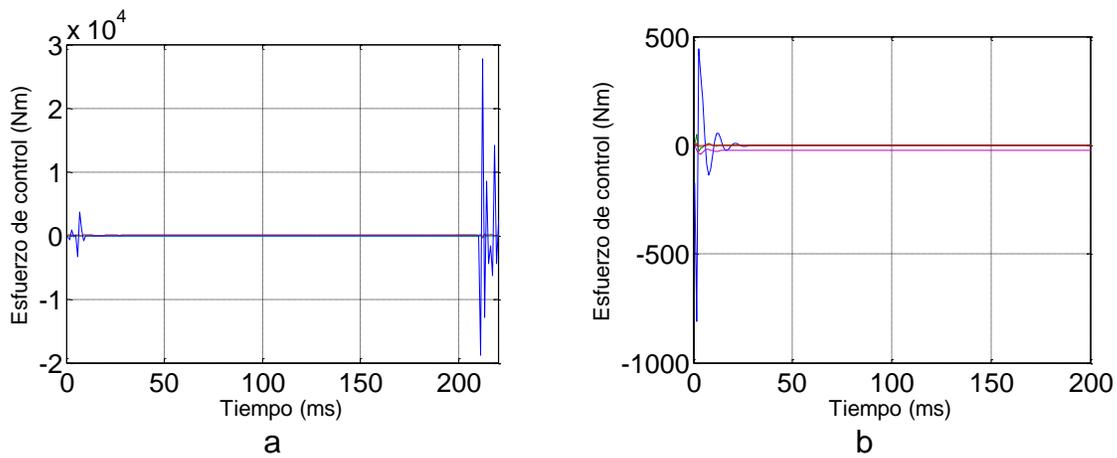


Figura 6.22: Par generado por los motores con incertidumbre del 20% 5 seg. (transitorio) a) CRS, b) CBI.

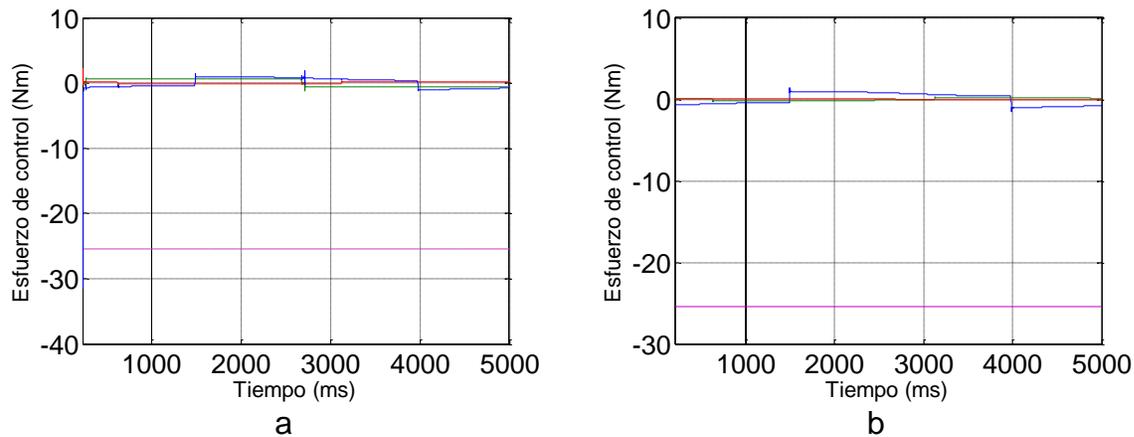


Figura 6.23: Par generado por los motores con incertidumbre del 20% 5 seg. (estacionario) a) CRS, b) CBI.

Para la trayectoria lineal, el esfuerzo de control a cambios de dirección bruscos se presenta en las figuras 6.24 a 6.26, donde se observan los resultados para el sistema sin incertidumbre y con una incertidumbre del 10 y 20% respectivamente. Igualmente para este caso el control CBI presenta menores esfuerzos de control lo cual refleja una ventaja adicional de esta estrategia de control frente al control CRS.

Tabla 6.4: Esfuerzo de control, CTC, CRS y CBI, para un tiempo de ejecución de 5 segundos.

% de incertid.	Esfuerzo de control CTC (Nm)			Esfuerzo de control CRS (Nm)			Esfuerzo de control CBI (Nm)		
	Circ. Trans.	Circ. Esta.	Lineal	Circ. Trans.	Circ. Esta.	Lineal	Circ. Trans.	Circ. Esta.	Lineal
0	540, -340	6, -4	60, -60	11300, -8600	1.6, -16	5500, -8700	500, -650	1.3, -17	300, -736
10	420, -300	5, -4	60, -60	40000, -23500	2.4, -21	430, -380	500, -740	1.3, -21	300, -74
-10	540, -410	8, -4	60, -60	Inestable		540, -580	510, -590	1.1, -14	300, -100
20	550, -570	5, -4	60, -60	27800, -18500	1.9, -25	430, -300	450, -810	1.5, -25	300, -74
-20	300, -220	10, -10	60, -60	Inestable			550, -520	0.9, -11	300, -100
40	270, -380	5, -9	60, -60	11200, -6800	1.7, -34	430, -180	450, -1100	1.7, -34	300, -69
-40	Inestable			Inestable			Inestable		
60	260, -330	4.5, -15	60, -60	16500, -17000	1.7, -45	2800, -2350	580, -1380	2, -45	300, -70
-60	Inestable			Inestable			Inestable		
100	250, -150	4.3, -29	60, -60	11500, -21600	2.3, -70	432, -70	1340, -3400	3, -70	300, -80

La tabla 6.5 muestra los valores pico (máximo y mínimo) de los esfuerzos de control de los controladores CTC, CRS y CBI para las trayectorias ejecutadas en 3 segundos.

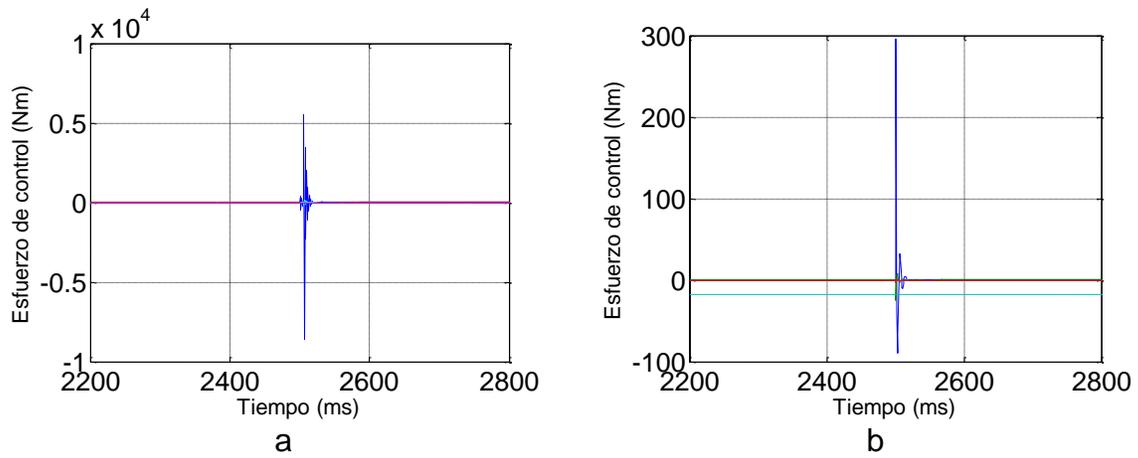


Figura 6.24: Par generado por los motores sin incertidumbre 5 seg. (cambio de dirección) a) CRS, b) CBI.

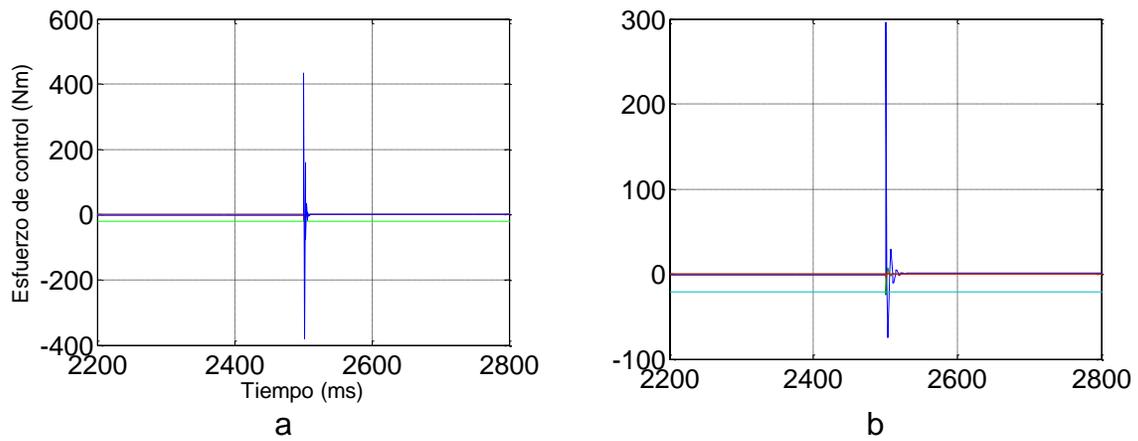


Figura 6.25: Par generado por los motores con incertidumbre del 10% 5 seg. (cambio de dirección) a) CRS, b) CBI.

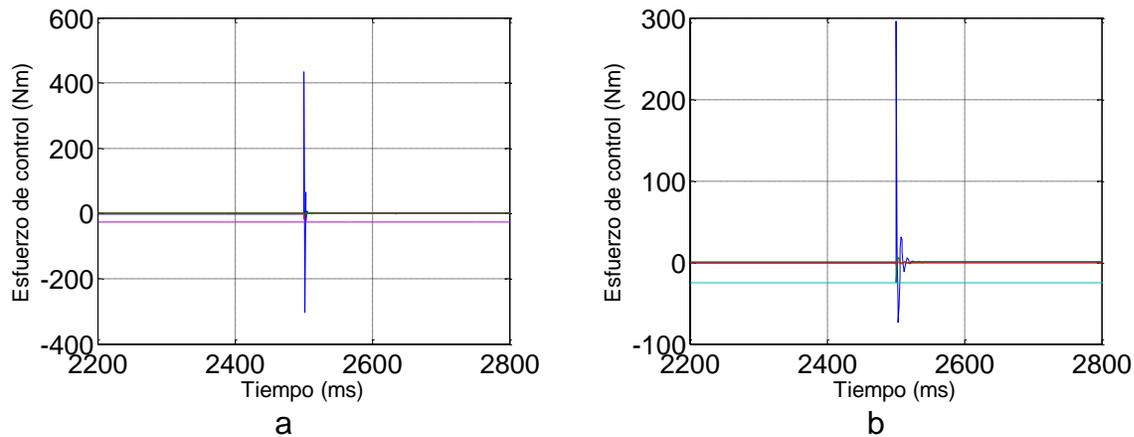


Figura 6.26: Par generado por los motores con incertidumbre del 20% 5 seg. (cambio de dirección) a) CRS, b) CBI.

Tabla 6.5: Esfuerzo de control, CTC, CRS y CBI para un tiempo de ejecución de 3 seg.

% de incertid.	Esfuerzo de control CTC (Nm)			Esfuerzo de control CRS (Nm)			Esfuerzo de control CBI (Nm)		
	Circ. Trans.	Circ. Esta.	Lineal	Circ. Trans.	Circ. Esta.	Lineal	Circ. Trans.	Circ. Esta.	Lineal
0	10000, -7500	9.5, -9.5	700, -900	9500, -7500	1.4, -17.5	420, -430	560, -940	1.7, -17	300, -90
10	9000, -7500	7, -7	750, -900	18000, -10000	1.8, -22	420, -380	500, -1000	2, -21	300, -75
-10	17000, -12000	13, -13	750, -900	Inestable		550, -580	620, -850	1.6, -14	300, -100
20	1900, -1900	7, -7	750, -900	65000, -45000	1.7, -25	420, -300	460, -1100	2.1, -25	300, -75
-20	4500, -3200	15, -15	750, -900	Inestable			710, -770	1.3, -11	300, -130
40	1500, -1500	6.5, -9	750, -900	20000, -15000	2.1, -35	420, -180	400, -1250	2.6, -34	300, -69
-40	Inestable			Inestable			Inestable		
60	6200, -6200	6.2, -15	750, -900	14000, -14000	2.2, -45	430, -90	400, -1510	3, -45	300, -70
-60	Inestable			Inestable			Inestable		
100	2200, -2200	5.2, -28	750, -900	20000, -26000	3.5, -70	430, -70	630, -2400	4.3, -70	300, -80

Observado los resultados resumidos en la tabla 6.4, es evidente que el control CBI muestra un menor esfuerzo de control que el control CRS para el estado transitorio de la trayectoria circular y para el cambio de dirección, mientras que para el estado estacionario el desempeño es muy similar al control CRS. Frente al control CTC, los esfuerzos de control que esta estrategia presenta son mejores que los presentados por el control CRS y CBI para el error de seguimiento de la trayectoria circular en estado transitorio y para el cambio de dirección.

Cuando se incrementa la velocidad de ejecución de la tarea, el control CRS presenta un mayor esfuerzo de control que el control CBI para la trayectoria circular en estado transitorio. Con respecto al cambio de dirección los esfuerzos de control de la estrategia CBI presentan un mejor desempeño que el control CTC y CRS, esto

permite evidenciar que el control CBI es una estrategia que permite controlar sistemas con alta incertidumbre sin presentar altos incrementos en los esfuerzos de control, lo cual no ocurre con el control *CTC* y *CRS*.

Capitulo 7

Conclusiones y trabajos futuros

Esta tesis se enfocó en el diseño y simulación de una ley de control robusta para un robot manipulador SCARA de 4 grados de libertad. La ley de control *backstepping* propuesta se basó en el modelo dinámico del manipulador, sin tener en cuenta dinámicas como la de los actuadores, las cuales son consideradas por otros autores (Lotfesar *et al.*, 2007; Li y Khajepour, 2005; Chen y Lin, 2004, Lotfesar *et al.*, 2003) para la aplicación de esta técnica de control, por otra parte se considero dentro del diseño de la ley de control incertidumbre paramétrica.

Las contribuciones de esta tesis fueron:

- Uno de los principales problemas en el control por modelo de referencia es que las dinámicas no modeladas, la incertidumbre paramétrica y el acoplamiento dinámico, afectan el desempeño del sistema de control. El principal aporte de este trabajo es la obtención de una ley de control que involucra la incertidumbre paramétrica y que presente un buen desempeño para el control de seguimiento.
- La ley de control propuesta se desarrolla mediante la metodología *backstepping*, la cual permite eliminar las no linealidades desestabilizantes y garantizar la estabilidad asintótica global del error.
- Otro aporte de esta tesis es el proveer de una nueva representación del modelo dinámico de un robot manipulador para lograr aplicar la técnica *backstepping*; las referencias estudiadas involucraban el modelo dinámico de los motores del robot para obtener así la representación del sistema y posteriormente desarrollar un control *backstepping*. En vista que en el proceso de identificación de los manipuladores se observó que los parámetros obtenidos en este proceso incluían los actuadores, el trabajo se enfocó en encontrar una representación sin tener en cuenta estas dinámicas y luego desarrollar un control *backstepping*. La ventaja que presenta la ley de control propuesta frente a la desarrollada por Lotfesar, Li, Chen, etc. (Lotfesar *et al.*,

2007; Li y Khajepour, 2005; Chen y Lin, 2004, Lotfesar *et al.*, 2003) es que este enfoque presenta buenos resultados aun sin el conocimiento del modelo de los actuadores, lo cual facilita el control de robots manipuladores ya que el proceso de identificar el modelo de los actuadores es dispendioso y requiere tener acceso al robot real, lo cual era una limitante en este trabajo.

- La técnica de control desarrollada presenta un buen desempeño frente al control robusto propuesto por Spong (Spong, 1992), en lo referente al error de seguimiento (en estado transitorio) y al esfuerzo de control. Es así como se logra una ley de control sencilla que permite estabilizar el sistema en presencia de incertidumbre paramétrica.
- Finalmente la ley de control *backstepping* con incertidumbre propuesta en esta tesis es una buena alternativa para el control de robots manipuladores cuando no se tiene un buen conocimiento del modelo dinámico del sistema, proveyendo sin embargo un buen desempeño en el seguimiento de trayectorias.

Como posibles trabajos futuros a desarrollar a partir del presente trabajo estarían:

El desarrollo de otras técnicas de control mediante la metodología *backstepping*, como puede ser el control adaptivo *backstepping*. Por otro parte un tema en el cual se podría explorar es en el desarrollo de observadores mediante la misma técnica.

Actualmente se encuentra en desarrollo un sistema que permite simular en 3D robusto manipuladores de 2 a 6 grados de libertad, obteniendo por medio de un sistema CAD los parámetros inerciales de los manipuladores para así lograr una simulación más cercana a la realidad. Una vez finalizado este proyecto se propone implementar las estrategias de control robusto de Spong y del control *backstepping* con incertidumbre y validar los resultados obtenidos en la simulación en Matlab.

Como etapa siguiente al control de robots en el sistema en 3D, se plantea el buscar alianzas con otras universidades del país que posean este tipo de manipuladores, con el fin de probar las diferentes estrategias de control estudiadas en esta tesis sobre un robot real, para el cual previamente se identificarían sus parámetros inerciales.

Referencias

- Adept Technology Inc. 2005. Disponible en: <http://www.adept.com/products/details.asp?pid=4>. Acceso junio de 2006.
- Alonge F., D'Ippolito F. y Raimondi F. "*Globally convergent adaptive and robust control of robotic manipulators for trajectory tracking*". Control Engineering Practice. 2004.
- Barrientos A., et al, "*Fundamentos de Robótica*", Mc Graw-Hill, España, 1998.
- Burkan R. y Uzmay I. "*Variable upper bounding approach for adaptive-robust control in robot control*". Journal of Intelligent and Robotic Systems. 2003.
- Burkan R. y Uzmay I. "*Logarithmic based robust control approach to parametric uncertainty for control of robot manipulators*". International Journal of Robust and Nonlinear Control. 2005.
- Casavola A. y Mosca E. " *H_∞ y H_2 simple controller for robotic applications*". International Journal of Control. 1992.
- Chan S. y Chen H. "*An efficient algorithm for identification of SCARA robot parameters including drive characteristics*". The 25th Annual Conference of the IEEE. Volume 2. 1999.
- Chen F. y Lin J. "*Nonlinear backstepping design of robot manipulators with velocity estimation feedback*", Proc. 5th Asian Control Conf., Melbourne, Australia, 2004.
- Chen C. "*Linear system theory and design*". Oxford University Press. 1999.
- Chun-Yi S, Yury S. y Sadik D. "*Hybrid integrator backstepping control of robotic manipulators driven by brushless DC motors*". IEEE/ASME Transactions on mechatronics. Vol. 1. No. 4. diciembre 1996.
- Chun-Yi S y Yury S. "*Backstepping based hybrid adaptive control of robot manipulators incorporating actuator dynamics*". International Journal of adaptive control and signal processing. Vol. 11. 1997.

Craig J.J. "*Introduction to robotics: mechanics and control*". Addison Wesley. USA. 1986.

Dorf R. C., "*International encyclopedia of robotics: applications and automation*". Jhon Wiley and Sons, Inc. 1988.

Goodwin G., Graebe.S, Salgado.M. "*Control system design*", Prentice Hall, 2000.

Härkegard O. "*Flight control design using backstepping*". Linköping Studies in Science and Technology. Thesis No 875. 2001.

Ishii C. "*Backstepping-Based Model Following Control with L2-Gain Performance for Robot Manipulators*". Research Reports Ashikaga Institute of Technology. Ashikaga. Japan. 1999.

Jagadeesh B. y Kurien K. "*Identification for the robust control of robot manipulators*". International Journal of Adaptive Control and Signal Processing. Vol. 15. Enero 2001.

Kemin Z. y Doyle J. "Esentials of roboust control". Prentice may 1998.

Khalil W., "*Nonlinear systems*". Prentice Hall, Nj, 1996.

Khalil W. y Dombé E., "*Modeling, identification and control of robots*". Bogan Page Science, London, 2002.

Krstic M., et al. "*Nonlinear and adaptive control design*". Wiley Inter Science, 1995.

Krstić M., Sun J.y Kokotović V. "*Robust control of nonlinear systems with inputunmodeled dynamics*". IEEE Transaction on Automatic Control. Vol. 41. No. 6. 1996.

Li G. y Khajepour A. "*Robust control of a hydraulically driven flexible arm using backstepping technique*". Journal of Sound and Vibration. 2005.

Li H. y Van den Bosch P. "*A robust disturbance-based control and its application*". International Journal of Control. 1993.

Li. W, Chang. X, Wahl. F.M. y Farrell, J. "*Design of an Enhanced Hybrid FUZZY P+ID controller for a mechanical manipulator*". IEEE Transactions on systems, man and cybernetics Part B: cybernetics, Vol.31, No.6, 2001.

Lin F. y Brandt R. "*An optimal control approach to robust control of robot manipulators*". IEEE Transactions on Robotics and Automation. Vol. 14. No. 1. Febrero 1998.

- Logothetis P. y Kieffer J. "On the Identification of Robot Dynamics without Acceleration Measurement". IEEE ICRA. 1996.
- Lotfesar A., Eghtesad M. y Mohseni M. "*Integrator backstepping control of a 5 DoF robot manipulator incorporating actuator dynamics*". Proceedings of 2003 IEEE Conference on Control Applications Vol. 2, Junio 2003.
- Lotfesar A., Eghtesad M. y Mohseni M. "*Application and comparison of passivity-based and integrator backstepping control methods for trajectory tracking of rigid-link robot manipulators incorporating motor dynamics*". International Journal of Robotics & Automation. Tomo 22, No. 3. Calgary: 2007.
- Lyapunov A. "*The general problem of the stability of motion*". Taylor & Francis. 1992. English translation of the original publication in Russian from 1892.
- Mnif F. "*A robust optimal controller for constrained robot manipulators*". Systems and Control Engineering. Vol. 218. Part I. 2004.
- Mnif F. "*A mixed optimal/robust control for robot manipulators*". Systems and Control Engineering. Vol. 218. Part I. 2004.
- Ollero. A. "*Robótica: Manipuladores y robots móviles*". Alfaomega, Barcelona, 2001.
- Paul R. "*Robot manipulator: mathematics, programming and control*". MIT Press, Cambridge. USA. 1981.
- Renteria A, y Rivas M. "*Robótica Industrial. Fundamentos y Aplicaciones*". McGraw-Hill. Madrid. 2000.
- Restrepo. P. P. "*Contribution á la modelization, identification et commande des robots a structures fermées : application au robot Acma SR400*". Université de Nantes, Ecole Centrale de Nantes. 1996.
- Rios M., Sira H., Zinober A. "*Output tracking control via adaptive input-output linearization: A backstepping approach*". Proceedings of the 34th Conference on Decision and Control. New Orleans, USA, 1995.
- Sage H. Mathelin M. y Ostertag E. "*Robust control of robot manipulators: a survey*". International Journal of Control. 1999.
- Sciavicco L. Siciliano B. "*Modeling and control of robots manipulators*". McGRAW-HILL. Singapore. 1996.
- Slotine J. "*Applied nonlinear control*". Prentice Hall. New Jersey. 1991.

Spong M.W. "On the robust control of robot manipulators." IEEE Transaction on Automatic Control. Vol. 37. No. 11. 1992.

Spong, M.W., "Motion Control of Robot Manipulators," in *Handbook of Control*, CRC Press.1996.

Sotang E. "A 'universal' construction of Artstein's theorem on nonlinear stabilization". *Sistems & Control Letters*. Vol. 13, pp. 117-123, 1989.

The MathWorks. 2006. "Signal constrain". Disponible en: http://www.mathworks.com/access/helpdesk/help/toolbox/sloptim/index.html?access/helpdesk/help/toolbox/sloptim/ug/signalconstraint.html&http://www.google.com.co/search?hl=es&rlz=1T4GFRC_esCO222CO222&sa=X&oi=spell&resnum=0&ct=result&cd=1&q=signal+constraint&spell=1. Acceso julio de 2007.

Vivas. A. "Control predictivo de un robot tipo SCARA". *Revista de Ingeniería Universidad de Tarapacá*, vol. 14 No. 2, pp. 135-145, 2006.

Willianson M. "Rhythmic Robot Arm Control Using Oscillators". *Conf. on Intelligent Robots and Systems*. Canada. 1998.

Willianson M. "Exploiting natural dynamics in robot control". *Fourteenth European Meeting on Cybernetics and Systems Research*. Austria. 1998.

Zhang Y. "Backstepping control of linear time-varying systems with known and unknown parameters". *IEEE Transactions on Automatic Control*. Vol. 48. No. 11. Nov. 2003

Zhou K. y Doyle J. "Essential of robust control". Prentice Hall. N.J. 1998.

Zhu G, Lee T., Ge s., "Tip Tracking Control of a Single-Link Flexible Robot: A Backstepping Approach" *Kluwer Academic Publishers Dynamics and Control*. Boston USA.1997.

Anexo A

Archivos para la simulación

A continuación se presentan los códigos que se desarrollaron en Matlab para la simulación del control del robot SCARA.

A.1 Función para simulación de las estrategias de control

1. Archivos correspondientes a las trayectorias circular y lineal los cuales se denominan circular_4gl.m y lineal_4gl.m respectivamente. Para ajustar el tiempo de ejecución de estas trayectorias se cambia la variable Tfinal al valor deseado y se ajustan las variables relacionadas, como son el tamaño de los vectores de cons3, y los ciclos if en la trayectoria lineal.

```
% Movimiento circular

Tfinal=5.0;
Tem=0.001;

% Cálculo del número de muestras %

nbech=(Tfinal/Tem)+1;
if ((round(nbech)-nbech) == 0)
    instant=[0:Tem:Tfinal]';
else
    nbech=nbech+1;
    instant=[0:Tem:Tfinal+Tem]';
end
t=0;

for h=1:1:nbech
    t=t+Tem;
    x1(h)=0.02*sin(2*pi*t/5);
```

```

        y1(h)=0.02*cos(2*pi*t/5);
    end
    x1=x1';
    y1=y1';
    %-----
    cons1= 0.4 + x1;
    cons2= 0.4 + y1;
    cons3=0.5*ones(5001,1);

```

% Movimiento lineal

```

Tfinal=5.0;
Tem=0.001;

% Cálculo del número de muestras %

nbech=(Tfinal/Tem)+1;
if ((round(nbech)-nbech) == 0)
    instant=[0:Tem:Tfinal]';
else
    nbech=nbech+1;
    instant=[0:Tem:Tfinal+Tem]';
end

instant = instant(1:5000,:);

t=0;
for h=1:1:2500
    t=t+Tem;
    x11(h)=t;
    y11(h)=t;
end

t=2.5;
for h=1:1:2500
    t=t+Tem;
    x22(h)=-t + 5;
    y22(h)= t;
end

xx = [x11 x22];
yy = [y11 y22];

xx = xx';
yy = yy';

```

```

cons1= 0.35 + 0.01*xx;
cons2= 0.35 + 0.01*yy;
cons3 = 0.4*ones(5000,1);

```

2. A continuación se presentan las funciones que calculan los modelos geométricos (mgd.m y mgi.m) y dinámicos (scara_directo.m y scara_inverso.m) del robot SCARA.

% Modelo geometrico directo del SCARA

```

function x = mgd(q1,q2,q3,q4)

d2 = 0.5;
d3 = 0.3;

TTT = [cos(q1+q2+q3) -sin(q1+q2+q3) 0 (d3*cos(q1+q2)+d2*cos(q1));
       sin(q1+q2+q3) cos(q1+q2+q3) 0 (d3*sin(q1+q2)+d2*sin(q1));
       0 0 1 q4;
       0 0 0 1];

xa = TTT(1,4);
ya = TTT(2,4);
za = TTT(3,4);

x = [xa ya za];

```

% Modelo geometrico inverso del SCARA

```

function q = mgi(x1,x2,x3)

d2 = 0.5;
d3 = 0.3;

C2 = (x1^2 + x2^2 - d2^2 - d3^2)/(2*d2*d3);
B1 = d2 + d3*C2;
q2 = atan2((-sqrt(1 - C2^2)),C2);
B2 = d3*sin(q2);
S1 = (B1*x2 - B2*x1)/(B1^2 + B2^2);
C1 = (B1*x1 + B2*x2)/(B1^2 + B2^2);

q1 = atan2(S1,C1);
q3 = atan2(0,-1) - q2 - q1;% sy, sx
q4 = x3;
q = [q1 q2 q3 q4];

```

%Modelo dinamico directo

```
function [sys, x0,str,ts] = f_scara(t,x,u,flag,QI)

switch flag,

% Initialization %
case 0,
    [sys,x0,str,ts]=mdlInitializeSizes(QI);

% Derivatives %
case 1,
    sys=mdlDerivatives(t,x,u);

% Outputs %
case 3,
    sys=mdlOutputs(t,x,u);

% Unhandled flags %
case { 2, 4, 9 },
    sys = [];

% Unexpected flags %
otherwise
    error(['Unhandled flag = ',num2str(flag)]);

end

% end f_scara

% mdlInitializeSizes
% Return the sizes, initial conditions, and sample times for the S-
function.

function [sys,x0,str,ts]=mdlInitializeSizes(QI)

sizes = simsizes;
sizes.NumContStates= 8;
sizes.NumDiscStates= 0;
sizes.NumOutputs= 8;
sizes.NumInputs= 4;
sizes.DirFeedthrough= 0;
sizes.NumSampleTimes= 1;

sys = simsizes(sizes);
```

```
% initialize the initial conditions

x0=[QI(1);QI(2);QI(3);QI(4);0;0;0;0];

% str is always an empty matrix

str = [];

% initialize the array of sample times

ts = [0 0];

% end mdlInitializeSizes

% mdlDerivatives
% Return the derivatives for the continuous states.

function sys=mdlDerivatives(t,x,u)

% Parametros del SCARA para la simulacion:

P=1.0;
G3=9.81*P;
M4=1.8*P;
d2=0.5*P;
d3=0.3*P;%
r4=0.2*P;%
ZZ1R=3.38*P;
ZZ2R=0.063*P;
ZZ3R=0.1*P;%
MX2R=2*0.121*P;
MX3R=0.2*P;%
MY3R=0.1*P;%
MY2=0.001*P;
FV1=0.1*P;
FV2=0.012*P;
FV3=0.01*P;
FV4=0.001*P;
FS1=0.57*P;
FS2=0.125*P;
FS3=0.05*P;
FS4=0.012*P;
IA2=0.045*P;%E0;
CZ1=0*P;
```

```
FX2=0*P;
FY2=0*P;
FX3=0*P;%
FY3=0*P;%
FZ3=0*P;%
CZ3=0*P;%
CZ4=0*P;%
CX3=0*P;%
CX4=0*P;%
CY3=0*P;%
CY4=0*P;%
FX4=0*P;%
FY4=0*P;%
FZ4=0*P;%
IA3=0.045*P;%E0;%
IA4=0.045;%

GAM1=u(1);
GAM2=u(2);
GAM3=u(3);
GAM4=u(4);

S1=sin(x(1));
C1=cos(x(1));
S2=sin(x(2));
C2=cos(x(2));
S3=sin(x(3));
C3=cos(x(3));

QP1=x(5);
QP2=x(6);
QP3=x(7);
QP4=x(8);

W32=QP1 + QP2;
JPR132=d2.*S2;
JPR232=C2.*d2;
W33=QP3 + W32;
JPR133=d3.*S3;
JPR233=C3.*d3;
JW31=QP1.*ZZ1R;
JW32=W32.*ZZ2R;
SW12=-(MX2R.*W32.^2);
SW22=-(MY2.*W32.^2);
LW12=-(C2.*d2.*QP1.^2);
```

LW22=d2.*QP1.^2.*S2;
JW33=W33.*ZZ3R;
SW13=-(MX3R.*W33.^2);
SW23=-(MY3R.*W33.^2);
LW13=-(C3.*d3.*W32.^2);
LW23=d3.*S3.*W32.^2;
JD4=1./(IA4 + M4);
JU64=JD4.*M4;
GW4=-FZ4 + GAM4 - FV4.*QP4 - FS4.*sign(QP4);
GK664=M4 - JU64.*M4;
VS64=GW4.*JU64;
AP64=FZ4 + VS64;
GX154=-(M4.*r4);
GX244=M4.*r4;
TKT114=-(GX154.*r4);
TKT514=-(M4.*r4);
TKT224=GX244.*r4;
TKT424=M4.*r4;
VBE13=-CX3 - CX4 + FY4.*r4;
VBE23=-CY3 - CY4 - FX4.*r4;
VBE33=-CZ3 - CZ4;
VBE43=-FX3 - FX4 - SW13;
VBE53=-FY3 - FY4 - SW23;
VBE63=-AP64 - FZ3;
JD3=1./(IA3 + ZZ3R);
JU33=JD3.*ZZ3R;
JU43=-(JD3.*MY3R);
JU53=JD3.*MX3R;
GW3=GAM3 - FV3.*QP3 + VBE33 - FS3.*sign(QP3);
GK333=ZZ3R - JU33.*ZZ3R;
GK343=-MY3R + JU33.*MY3R;
GK353=MX3R - JU33.*MX3R;
GK433=-MY3R - JU43.*ZZ3R;
GK443=M4 + JU43.*MY3R;
GK453=-(JU43.*MX3R);
GK533=MX3R - JU53.*ZZ3R;
GK543=JU53.*MY3R;
GK553=M4 - JU53.*MX3R;
NG13=LW23.*TKT514;
NG23=LW13.*TKT424;
NG33=GK343.*LW13 + GK353.*LW23;
NG43=GK443.*LW13 + GK453.*LW23;
NG53=GK543.*LW13 + GK553.*LW23;
VS33=GW3.*JU33 + NG33;
VS43=GW3.*JU43 + NG43;

VS53=GW3.*JU53 + NG53;
AP13=NG13 - VBE13;
AP23=NG23 - VBE23;
AP33=-VBE33 + VS33;
AP43=-VBE43 + VS43;
AP53=-VBE53 + VS53;
GX113=C3.*TKT114;
GX123=-(S3.*TKT224);
GX143=-(S3.*TKT424);
GX153=C3.*TKT514;
GX163=C3.*MY3R + MX3R.*S3;
GX213=-(d3.*MY3R) + S3.*TKT114;
GX223=d3.*MX3R + C3.*TKT224;
GX243=C3.*TKT424;
GX253=S3.*TKT514;
GX263=-(d3.*GK664) - C3.*MX3R + MY3R.*S3;
GX313=JPR233.*TKT514;
GX323=JPR133.*TKT424;
GX333=GK333 + GK433.*JPR133 + GK533.*JPR233;
GX343=GK343 + GK443.*JPR133 + GK543.*JPR233;
GX353=GK353 + GK453.*JPR133 + GK553.*JPR233;
GX413=-(S3.*TKT514);
GX423=C3.*TKT424;
GX433=C3.*GK433 - GK533.*S3;
GX443=C3.*GK443 - GK543.*S3;
GX453=C3.*GK453 - GK553.*S3;
GX513=C3.*TKT514;
GX523=S3.*TKT424;
GX533=C3.*GK533 + GK433.*S3;
GX543=C3.*GK543 + GK443.*S3;
GX553=C3.*GK553 + GK453.*S3;
TKT113=C3.*GX113 - GX123.*S3;
TKT213=C3.*GX213 - GX223.*S3;
TKT313=C3.*GX313 - GX323.*S3;
TKT413=C3.*GX413 - GX423.*S3;
TKT513=C3.*GX513 - GX523.*S3;
TKT613=C3.*MY3R + MX3R.*S3;
TKT223=C3.*GX223 - d3.*GX263 + GX213.*S3;
TKT323=C3.*GX323 + GX313.*S3;
TKT423=C3.*GX423 + GX413.*S3;
TKT523=C3.*GX523 + GX513.*S3;
TKT623=-(d3.*GK664) - C3.*MX3R + MY3R.*S3;
TKT333=GX333 + GX343.*JPR133 + GX353.*JPR233;
TKT433=GX433 + GX443.*JPR133 + GX453.*JPR233;
TKT533=GX533 + GX543.*JPR133 + GX553.*JPR233;

TKT443=C3.*GX443 - GX453.*S3;
TKT543=C3.*GX543 - GX553.*S3;
TKT553=C3.*GX553 + GX543.*S3;
MJE612=MY2 + TKT613;
MJE622=-MX2R + TKT623;
MJE332=TKT333 + ZZ2R;
MJE432=-MY2 + TKT433;
MJE532=MX2R + TKT533;
VBE12=-(AP13.*C3) + AP23.*S3;
VBE22=-(AP23.*C3) - AP13.*S3 - d3.*VBE63;
VBE32=-AP33 - AP43.*JPR133 - AP53.*JPR233;
VBE42=-(AP43.*C3) + AP53.*S3 - SW12;
VBE52=-(AP53.*C3) - AP43.*S3 - SW22;
JD2=1./(IA2 + MJE332);
JU12=JD2.*TKT313;
JU22=JD2.*TKT323;
JU32=JD2.*MJE332;
JU42=JD2.*MJE432;
JU52=JD2.*MJE532;
GW2=GAM2 - FV2.*QP2 + VBE32 - FS2.*sign(QP2);
GK112=TKT113 - JU12.*TKT313;
GK122=TKT213 - JU12.*TKT323;
GK132=-(JU12.*MJE332) + TKT313;
GK142=-(JU12.*MJE432) + TKT413;
GK152=-(JU12.*MJE532) + TKT513;
GK212=TKT213 - JU22.*TKT313;
GK222=TKT223 - JU22.*TKT323;
GK232=-(JU22.*MJE332) + TKT323;
GK242=-(JU22.*MJE432) + TKT423;
GK252=-(JU22.*MJE532) + TKT523;
GK312=TKT313 - JU32.*TKT313;
GK322=TKT323 - JU32.*TKT323;
GK332=MJE332 - JU32.*MJE332;
GK342=MJE432 - JU32.*MJE432;
GK352=MJE532 - JU32.*MJE532;
GK412=-(JU42.*TKT313) + TKT413;
GK422=-(JU42.*TKT323) + TKT423;
GK432=-(JU42.*MJE332) + MJE432;
GK442=-(JU42.*MJE432) + TKT443;
GK452=-(JU42.*MJE532) + TKT543;
GK512=-(JU52.*TKT313) + TKT513;
GK522=-(JU52.*TKT323) + TKT523;
GK532=-(JU52.*MJE332) + MJE532;
GK542=-(JU52.*MJE432) + TKT543;
GK552=-(JU52.*MJE532) + TKT553;

NG12=GK142.*LW12 + GK152.*LW22;
NG22=GK242.*LW12 + GK252.*LW22;
NG32=GK342.*LW12 + GK352.*LW22;
NG42=GK442.*LW12 + GK452.*LW22;
NG52=GK542.*LW12 + GK552.*LW22;
VS12=GW2.*JU12 + NG12;
VS22=GW2.*JU22 + NG22;
VS32=GW2.*JU32 + NG32;
VS42=GW2.*JU42 + NG42;
VS52=GW2.*JU52 + NG52;
AP12=-VBE12 + VS12;
AP22=-VBE22 + VS22;
AP32=-VBE32 + VS32;
AP42=-VBE42 + VS42;
AP52=-VBE52 + VS52;
GX112=C2.*GK112 - GK212.*S2;
GX122=C2.*GK122 - GK222.*S2;
GX132=C2.*GK132 - GK232.*S2;
GX142=C2.*GK142 - GK242.*S2;
GX152=C2.*GK152 - GK252.*S2;
GX162=C2.*MJE612 - MJE622.*S2;
GX212=C2.*GK212 - d2.*MJE612 + GK112.*S2;
GX222=C2.*GK222 - d2.*MJE622 + GK122.*S2;
GX232=C2.*GK232 + GK132.*S2;
GX242=C2.*GK242 + GK142.*S2;
GX252=C2.*GK252 + GK152.*S2;
GX262=- (d2.*GK664) + C2.*MJE622 + MJE612.*S2;
GX312=GK312 + GK412.*JPR132 + GK512.*JPR232;
GX322=GK322 + GK422.*JPR132 + GK522.*JPR232;
GX332=GK332 + GK432.*JPR132 + GK532.*JPR232;
GX342=GK342 + GK442.*JPR132 + GK542.*JPR232;
GX352=GK352 + GK452.*JPR132 + GK552.*JPR232;
GX412=C2.*GK412 - GK512.*S2;
GX422=C2.*GK422 - GK522.*S2;
GX432=C2.*GK432 - GK532.*S2;
GX442=C2.*GK442 - GK542.*S2;
GX452=C2.*GK452 - GK552.*S2;
GX512=C2.*GK512 + GK412.*S2;
GX522=C2.*GK522 + GK422.*S2;
GX532=C2.*GK532 + GK432.*S2;
GX542=C2.*GK542 + GK442.*S2;
GX552=C2.*GK552 + GK452.*S2;
TKT112=C2.*GX112 - GX122.*S2;
TKT212=C2.*GX212 - GX222.*S2;
TKT312=C2.*GX312 - GX322.*S2;

$TKT412=C2.*GX412 - GX422.*S2;$
 $TKT512=C2.*GX512 - GX522.*S2;$
 $TKT612=C2.*MJE612 - MJE622.*S2;$
 $TKT222=C2.*GX222 - d2.*GX262 + GX212.*S2;$
 $TKT322=C2.*GX322 + GX312.*S2;$
 $TKT422=C2.*GX422 + GX412.*S2;$
 $TKT522=C2.*GX522 + GX512.*S2;$
 $TKT622=-(d2.*GK664) + C2.*MJE622 + MJE612.*S2;$
 $TKT332=GX332 + GX342.*JPR132 + GX352.*JPR232;$
 $TKT432=GX432 + GX442.*JPR132 + GX452.*JPR232;$
 $TKT532=GX532 + GX542.*JPR132 + GX552.*JPR232;$
 $TKT442=C2.*GX442 - GX452.*S2;$
 $TKT542=C2.*GX542 - GX552.*S2;$
 $TKT552=C2.*GX552 + GX542.*S2;$
 $MJE331=TKT332 + ZZ1R;$
 $VBE11=-(AP12.*C2) + AP22.*S2;$
 $VBE21=-(AP22.*C2) - AP12.*S2 - d2.*VBE63;$
 $VBE31=-AP32 - AP42.*JPR132 - AP52.*JPR232;$
 $VBE41=-(AP42.*C2) + AP52.*S2;$
 $VBE51=-(AP52.*C2) - AP42.*S2;$
 $JD1=1./MJE331;$
 $JU11=JD1.*TKT312;$
 $JU21=JD1.*TKT322;$
 $JU31=JD1.*MJE331;$
 $JU41=JD1.*TKT432;$
 $JU51=JD1.*TKT532;$
 $GW1=GAM1 - FV1.*QP1 + VBE31 - FS1.*sign(QP1);$
 $QDP1=GW1.*JD1;$
 $VR42=LW12 + JPR132.*QDP1;$
 $VR52=LW22 + JPR232.*QDP1;$
 $GU2=JU32.*QDP1 + JU42.*VR42 + JU52.*VR52;$
 $QDP2=-GU2 + GW2.*JD2;$
 $WP32=QDP1 + QDP2;$
 $VR43=LW13 + C3.*VR42 + S3.*VR52 + JPR133.*WP32;$
 $VR53=LW23 - S3.*VR42 + C3.*VR52 + JPR233.*WP32;$
 $GU3=JU43.*VR43 + JU53.*VR53 + JU33.*WP32;$
 $QDP3=-GU3 + GW3.*JD3;$
 $WP33=QDP3 + WP32;$
 $GU4=-(G3.*JU64);$
 $QDP4=-GU4 + GW4.*JD4;$
 $sys(1) = x(5) ;$
 $sys(2) = x(6) ;$
 $sys(3) = x(7) ;$
 $sys(4) = x(8) ;$
 $sys(5) = QDP1;$

```

sys(6) = QDP2;
sys(7) = QDP3;
sys(8) = QDP4;

% end mdlDerivatives

% mdlOutputs
% Return the block outputs.

function sys=mdlOutputs(t,x,u)

    sys(1) = x(1);
    sys(2) = x(2);
    sys(3) = x(3);
    sys(4) = x(4);
    sys(5) = x(5);
    sys(6) = x(6);
    sys(7) = x(7);
    sys(8) = x(8);

% end mdlOutputs

```

%modelo dinámico inverso

```

function GAM = scara_inverso(position1,position2,position3,position4,
                             vitesse1,vitesse2,vitesse3,vitesse4,
                             w1,w2,w3,w4)

P=1.0; % indice para afectar los parametros por incertidumbre
G3=9.81*P;
M4=1.8*P;
d2=0.5*P;
d3=0.3*P;
ZZ1R=3.38*P;
ZZ2R=0.063*P;
ZZ3R=0.1*P;
MX2R=2*0.121*P;
MX3R=0.2*P;
MY3R=0.1*P;
MY2=0.001*P;
FV1=0.1*P;
FV2=0.012*P;
FV3=FV1*P;
FV4=FV2*P;
FS1=0.57*P;

```

```
FS2=0.125*P;  
FS3=FS1*P;  
FS4=FS2*P;  
IA2=0;  
CZ1=0;  
FX2=0;  
FY2=0;  
FX3=0;  
FY3=0;  
FZ3=0;  
CZ3=0;  
CZ4=0;  
FX4=0;  
FY4=0;  
FZ4=0;  
IA3=0;  
IA4=0.045;  
  
t2=position2;  
t3=position3;  
QP1=vitesse1;  
QP2=vitesse2;  
QP3=vitesse3;  
QP4=vitesse4;  
QDP1=w1;  
QDP2=w2;  
QDP3=w3;  
QDP4=w4;  
  
S2=sin(t2);  
C2=cos(t2);  
S3=sin(t3);  
C3=cos(t3);  
  
DV331=-QP1.^2;  
No31=QDP1.*ZZ1R;  
W32=QP1 + QP2;  
WP32=QDP1 + QDP2;  
DV332=-W32.^2;  
VSP12=d2.*DV331;  
VSP22=d2.*QDP1;  
VP12=C2.*VSP12 + S2.*VSP22;  
VP22=-(S2.*VSP12) + C2.*VSP22;  
F12=DV332.*MX2R - MY2.*WP32;  
F22=DV332.*MY2 + MX2R.*WP32;
```

```

No32=WP32.*ZZ2R;
W33=QP3 + W32;
WP33=QDP3 + WP32;
DV333=-W33.^2;
VSP13=d3.*DV332 + VP12;
VSP23=VP22 + d3.*WP32;
VP13=C3.*VSP13 + S3.*VSP23;
VP23=-(S3.*VSP13) + C3.*VSP23;
F13=DV333.*MX3R - MY3R.*WP33;
F23=DV333.*MY3R + MX3R.*WP33;
No33=WP33.*ZZ3R;
VP34=-G3 + QDP4;
F14=M4.*VP13;
F24=M4.*VP23;
F34=M4.*VP34;
E14=F14 + FX4;
E24=F24 + FY4;
E34=F34 + FZ4;
E13=E14 + F13 + FX3;
E23=E24 + F23 + FY3;
N33=CZ3 + CZ4 + No33 - MY3R.*VP13 + MX3R.*VP23;
FDI13=C3.*E13 - E23.*S3;
FDI23=C3.*E23 + E13.*S3;
E12=F12 + FDI13;
E22=F22 + FDI23;
N32=d3.*FDI23 + N33 + No32 - MY2.*VP12 + MX2R.*VP22;
FDI22=C2.*E22 + E12.*S2;
N31=d2.*FDI22 + N32 + No31;
GAM1=N31 + FV1.*QP1 + FS1.*sign(QP1);
GAM2=N32 + IA2.*QDP2 + FV2.*QP2 + FS2.*sign(QP2);
GAM3=N33 + IA3.*QDP3 + FV3.*QP3 + FS3.*sign(QP3);
GAM4=E34 + IA4.*QDP4 + FV4.*QP4 + FS4.*sign(QP4);

GAM(1) = GAM1;
GAM(2) = GAM2;
GAM(3) = GAM3;
GAM(4) = GAM4;

```

3. Programas para la simulación del control PID y CTC

% Inicializacion control PID SCARA

```

clear all
close all
clc

```

```
% Condición inicial del efecto final (trayectoria lineal)

%QI=[1.3979;-1.8930;3.6367;0.5000];

% Condición inicial del efecto final (trayectoria circular)

QI=[1.3534;-1.5828;3.3710; 0.5000];

Tech = 0.001;
Tech_com = Tech;

% selección de trayectoria

%circular;
lineal;

% constantes de las matrices del controlador
KP1=220000;
KV1=150;
KI1=1000;

KP2=280000;
KV2=170;
KI2=1000;

KP3=80000;
KV3=120;
KI3=1000;

KP4=200000;
KV4=250;
KI4=1000;
```

% Inicializacion control CTC SCARA:

```
clear all
close all
clc

% Condición inicial del efecto final (trayectoria lineal)

%QI=[1.3979;-1.8930;3.6367;0.5000];

% Condición inicial del efecto final (trayectoria circular)

QI=[1.3534;-1.5828;3.3710; 0.5000];
```

```

Tem = 0.001;

%selección de trayectoria
circular;
%lineal;

KP1=320000;
KV1=1200;

KP2=280000;
KV2=850;
KP3=200000;
KV3=1200;

KP4=120000;
KV4=700;

```

4. Programa necesarios para la simulación del control robusto de Spong.

```

% Calculo Matriz de funciones de coordenadas generalizadas Y

function y = Y_4gl(qr1dd,qr2dd,qr3dd,qr4dd,q1,q2,q3)

g = 9.8;
D2 = 0.5;
D3 = 0.3;

y11 = qr1dd;
y12 = qr1dd + 2*qr2dd;
y13 = qr1dd + 2*qr2dd + qr3dd;
y14 = qr1dd + qr2dd + qr3dd;
y15 = 2*cos(q2)*D2*qr1dd + 2*cos(q2)*D2*qr2dd;
y16 = (-sin(q2)*sin(q3)*D2+cos(q3)*cos(q2)*D2+
        D3*cos(q3))*(2*qr1dd + 2*qr2dd + qr3dd)+
        2*D3*cos(q3)*qr2dd;
y17 = -2*D2*sin(q2)*qr1dd-2*sin(q2)*D2*qr2dd;
y18 = -(sin(q2)*sin(q3)*D2 + cos(q2)*sin(q3)*D2 +
        D3*sin(q3))*(2*qr1dd + 2*qr2dd + qr3dd) -
        2*D3*sin(q3)*qr2dd;
y19 = 0;
y110 = 0;
y111 = 0;

y21 = 0;

```

```

y22 = 2*qr1dd + qr2dd;
y23 = 2*qr1dd + qr2dd + qr3dd;
y24 = qr1dd + qr2dd + qr3dd;
y25 = 2*cos(q2)*D2*qr1dd;
y26 = 2*(-sin(q2)*sin(q3)*D2 + cos(q3)*cos(q2)*D2 +
        D3*cos(q3))*qr1dd + D3*cos(q3)*
        (2*qr1dd + 2*qr2dd + qr3dd);
y27 = -2*sin(q2)*D2*qr1dd;
y28 = -2*D3*sin(q3)*(qr1dd + qr2dd + qr3dd) - (sin(q2)*cos(q3)*D2 +
        D2*cos(q2)*sin(q3) + D3*sin(q3));
y29 = qr2dd;
y210 = 0;
y211 = 0;

y31 = 0;
y32 = 0;
y33 = qr1dd + qr2dd + qr3dd;
y34 = qr1dd + qr2dd + qr3dd;
y35 = 0;
y36 = (-sin(q2)*sin(q3)*D2 + cos(q3)*cos(q2)*D2 + D3*cos(q3))*qr1dd
        + D3*cos(q3)*qr2dd;
y37 = 0;
y38 = -(sin(q2)*cos(q3)*D2 + cos(q2)*sin(q3)*D2 + D3*cos(q3))*qr1dd
        - 2*D3*sin(q3)*qr2dd;
y39 = 0;
y310 = qr3dd;
y311 = 0;

y41 = 0;
y42 = 0;
y43 = 0;
y44 = 0;
y45 = 0;
y46 = 0;
y47 = 0;
y48 = 0;
y49 = 0;
y410 = 0;
y411 = qr4dd;

y = [y11 y12 y13 y14 y15 y16 y17 y18 y19 y110 y111;
      y21 y22 y23 y24 y25 y26 y27 y28 y29 y210 y211;
      y31 y32 y33 y34 y35 y36 y37 y38 y39 y310 y311;
      y41 y42 y43 y44 y45 y46 y47 y48 y49 y410 y411];

```

% Calculo de la ley de control Spong

```

function tau =
    TAU_4gl(qr1dd,qr2dd,qr3dd,qr4dd,q1,q2,q3,a1,a2,a3,a4)
epsilon=1000;
rho=0.2;

g = 9.8;
D2 = 0.5;
D3 = 0.3;

y11 = qr1dd;
y12 = qr1dd + 2*qr2dd;
y13 = qr1dd + 2*qr2dd + qr3dd;
y14 = qr1dd + qr2dd + qr3dd;
y15 = 2*cos(q2)*D2*qr1dd + 2*cos(q2)*D2*qr2dd;
y16 = (-sin(q2)*sin(q3)*D2+cos(q3)*cos(q2)*D2+
    D3*cos(q3))*(2*qr1dd + 2*qr2dd + qr3dd)+
    2*D3*cos(q3)*qr2dd;
y17 = -2*D2*sin(q2)*qr1dd-2*sin(q2)*D2*qr2dd;
y18 = -(sin(q2)*sin(q3)*D2 + cos(q2)*sin(q3)*D2 +
    D3*sin(q3))*(2*qr1dd + 2*qr2dd + qr3dd) -
    2*D3*sin(q3)*qr2dd;
y19 = 0;
y110 = 0;
y111 = 0;

y21 = 0;
y22 = 2*qr1dd + qr2dd;
y23 = 2*qr1dd + qr2dd + qr3dd;
y24 = qr1dd + qr2dd + qr3dd;
y25 = 2*cos(q2)*D2*qr1dd;
y26 = 2*(-sin(q2)*sin(q3)*D2 + cos(q3)*cos(q2)*D2 +
    D3*cos(q3))*qr1dd + D3*cos(q3)*
    (2*qr1dd + 2*qr2dd + qr3dd);
y27 = -2*sin(q2)*D2*qr1dd;
y28 = -2*D3*sin(q3)*(qr1dd + qr2dd + qr3dd) - (sin(q2)*cos(q3)*D2 +
    D2*cos(q2)*sin(q3) + D3*sin(q3));
y29 = qr2dd;
y210 = 0;
y211 = 0;

y31 = 0;
y32 = 0;
y33 = qr1dd + qr2dd + qr3dd;

```

```

y34 = qr1dd + qr2dd + qr3dd;
y35 = 0;
y36 = (-sin(q2)*sin(q3)*D2 + cos(q3)*cos(q2)*D2 + D3*cos(q3))*qr1dd
      + D3*cos(q3)*qr2dd;
y37 = 0;
y38 = -(sin(q2)*cos(q3)*D2 + cos(q2)*sin(q3)*D2 + D3*cos(q3))*qr1dd
      - 2*D3*sin(q3)*qr2dd;
y39 = 0;
y310 = qr3dd;
y311 = 0;

```

```

y41 = 0;
y42 = 0;
y43 = 0;
y44 = 0;
y45 = 0;
y46 = 0;
y47 = 0;
y48 = 0;
y49 = 0;
y410 = 0;
y411 = qr4dd;

```

```

y = [y11 y12 y13 y14 y15 y16 y17 y18 y19 y110 y111;
     y21 y22 y23 y24 y25 y26 y27 y28 y29 y210 y211;
     y31 y32 y33 y34 y35 y36 y37 y38 y39 y310 y311;
     y41 y42 y43 y44 y45 y46 y47 y48 y49 y410 y411];

```

```

a = [a1;a2;a3;a4];

```

```

if norm(y'*a) > epsilon
    tau = -rho*y'*a/norm(y'*a);
else
    tau = -rho*y'*a/epsilon;
end

```

% diferencia (obtener error de seguimiento)

```

function dif = diferencia(a1,a2,a3,a4)

d=abs(sqrt((a1 - a3)^2 + (a2 - a4)^2));

Dif = d;

```

```

% Inicialización de control Spong
clear all
close all
clc

Tem=0.01;

% selección de trayectoria

%lineal_4gl;
Circular_4gl;

% Condición inicial del efecto final (trayectoria lineal)
%Ql=[1.3979;-1.8930;3.6367;0.5000];

% Condición inicial del efecto final (trayectoria circular)
Ql=[1.3534;-1.5828;3.3710; 0.5000];

% valores nominales de los parámetros del robot
PI0=[3.38 0.063 0.1 0 0.242 0.2 0.001 0.1 0 0 0.045]';

% Definición de matrices Delta y K

Delta = diag([1550 550 70 1500]);
K      = diag([1200 200 200 1500]);

```

5. Programas para simulación del control *backstepping*.

```

% calculo de la matriz M

function M = Matriz_M_4gl(q1,q2,q3,q4)

% Modelo geometrico directo del SCARA de cuatro ejes:

P=1.0;
G3=9.81*P;
M4=1.8*P;
D2=0.5*P;
D3=0.3*P;%
r4=0.2*P;%
ZZ1R=3.38*P;
ZZ2R=0.063*P;

```

```

ZZ3R=0.1*P;%
MX2R=2*0.121*P;
MX3R=0.2*P;%
MY3R=0.1*P;%
MY2=0.001*P;
FV1=0.1*P;
FV2=0.012*P;
FV3=0.01*P;
FV4=0.001*P;
FS1=0.57*P;
FS2=0.125*P;
FS3=0.05*P;
FS4=0.012*P;
Ia2=0.045*P;
CZ1=0*P;
FX2=0*P;
FY2=0*P;
FX3=0*P;%
FY3=0*P;%
FZ3=0*P;%
CZ3=0*P;%
CZ4=0*P;%
CX3=0*P;%
CX4=0*P;%
CY3=0*P;%
CY4=0*P;%
FX4=0*P;%
FY4=0*P;%
FZ4=0*P;%
Ia3=0.045*P;%E0;%
Ia4=0.045;%

S1=sin(q1);
C1=cos(q1);
S2=sin(q2);
C2=cos(q2);
S3=sin(q3);
C3=cos(q3);
C32=cos(q3+q2);
S32=sin(q3+q2);

M=[ZZ1R+ZZ2R+2*MX2R*C2*D2-2*MY2*S2*D2+ZZ3R+
  2*MX3R*D2*C32+2*MX3R*C3*D3-2*MY3R*D2*S32-
  2*MY3R*S3*D3+M4*D2^2+2*M4*S2*D2*D3+M4*D3^2,
  ZZ2R+MX2R*C2*D2-MY2*S2*D2+ZZ3R+MX3R*D2*C32

```

```

+2*MX3R*C3*D3-MY3R*D2*S32-2*MY3R*S3*D3+M4*C2*D2*D3+
M4*D3^2,
ZZ3R+MX3R*D2*C32+MX3R*C3*D3-MY3R*D2*S32-
MY3R*S3*D3, 0
ZZ2R+MX2R*C2*D2-MY2*S2*D2+ZZ3R+MX3R*D2*C32+
2*MX3R*C3*D3-MY3R*D2*S32-2*MY3R*S3*D3+
M4*C2*D2*D3+M4*D3^2,
ZZ2R+ZZ3R+2*MX3R*C3*D3-2*MY3R*S3*D3+M4*D3^2+Ia2,
ZZ3R+MX3R*C3*D3-MY3R*S3*D3,
0, ZZ3R+MX3R*D2*C32+MX3R*C3*D3-
MY3R*D2*S32-MY3R*S3*D3, ZZ3R+MX3R*C3*D3-
MY3R*S3*D3, ZZ3R+Ia3, 0, 0, 0, 0, M4+Ia4];

```

% Inicializacion control BACKSTEPPING robot SCARA:

```

clear all
close all
clc

```

```

Tem=0.01;

```

```

% Trayectorias

```

```

%lineal_4gl;

```

```

Circular_4gl;

```

```

% condiciones iniciales del efector final para las trayectorias

```

```

%QI = [1.3979;-1.8930;3.6367;0.5000]; % lineal

```

```

QI=[1.3534;-1.5828;3.3710; 0.5000]; %circular

```

```

% definición de las variables del controlador

```

```

delta =0.2; % valor de incertidumbre

```

```

K = 900*diag([1 1 1 1]);

```

6. Programa de calculo de máximos de la matriz **M**

% Maximos de la diagonal

```

% Computing Ajmax

```

```

clc

```

```

% parameters

```

```

G3=9.81;

```

```

M4=1.8;

```

```

D2=0.5;

```

```

D3=0.3;%

```

```

r4=0.2;

```

```

ZZ4=0;%

```

```
ZZ1R=3.38;
ZZ2R=0.063;
ZZ3R=0.1;%
MX2R=2*0.121;
MX3R=0.2;%
MY3R=0.1;%
MY2=0.001;
FV1=0.1;
FV2=0.012;
FV3=0.01;
FV4=0.001;
FS1=0.57;
FS2=0.125;
FS3=0.05;
FS4=0.012;
IA2=0;
CZ1=0;
FX2=0;
FY2=0;
FX3=0;%
FY3=0;%
FZ3=0;%
CZ3=0;%
CZ4=0;%
CX3=0;%
CX4=0;%
CY3=0;%
CY4=0;%
FX4=0;%
FY4=0;%
FZ4=0;%
IA3=0;%
IA4=0.045;%

S1=sin(Pos(:,1));
C1=cos(Pos(:,1));
S2=sin(Pos(:,2));
C2=cos(Pos(:,2));
S3=sin(Pos(:,3));
C3=cos(Pos(:,3));

% A11=[];
% A22=[];
% A33=[];
% A44=[];
```

```
% % Calculo de los Ajj
```

```
for i=1:length(Pos)
```

```
    A11(i)=ZZ1R+ZZ2R+(2*MX2R*C2(i)*D2)+(2*MY2*D2*S2(i))+((2*MX3R  
    )*(-S2(i)*S3(i)*D2)+(C3(i)*C2(i)*D2)+(D3*C3(i))))+(2*MY3R)*((-  
    S2(i)*C3(i)*D2)-(D2*C2(i)*S3(i))-(D3*S3(i))))+ZZ4+ZZ3R;
```

```
    A22(i)=IA2+ZZ2R+ZZ3R+(2*MX3R*D3*C3(i))-(2*MY3R*D3*S3(i))+ZZ4;
```

```
    A33(i)=IA3+ZZ3R+ZZ4;
```

```
    A44(i)=IA4;
```

```
end
```

```
A11_max=max(A11)
```

```
A22_max=max(A22)
```

```
A33_max=max(A33)
```

```
A44_max=max(A44)
```

Anexo B

Sintonización de control PID y CTC por el método de frecuencia de resonancia

En este anexo se presenta la sintonización por frecuencia de oscilación, técnica que no posee una amplia referencia bibliográfica, pero que se adicionó a este trabajo para verificar el procedimiento y que es referenciado en Khalil (Khalil y Dombre, 2002); Así mismo se presenta la sintonización por medio del bloque de *Simulink Signal Constrain*, el cual se utilizó para la sintonización de las estrategias de control desarrolladas en esta tesis.

B.1 Frecuencia de resonancia del robot

Para determinar la frecuencia de resonancia de cada uno de las articulaciones que componen el robot, se excita el sistema en lazo cerrado (figura B.1) con una señal senoidal de amplitud constante y su frecuencia se incrementa con el tiempo (Willianson, 1998), como se muestra en la figura B.1.

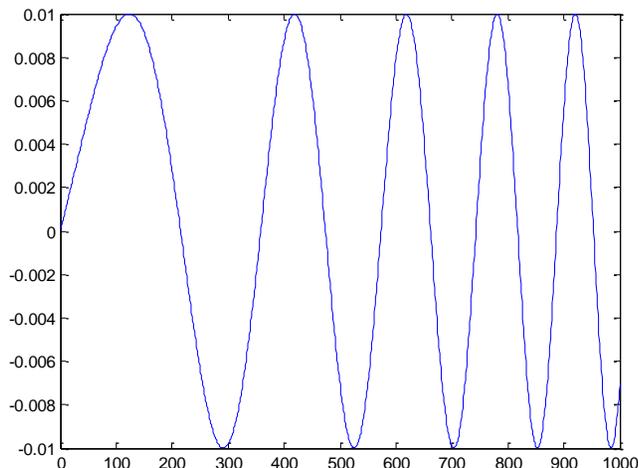


Figura B.1: Señal de entrada para determinar w_r .

La frecuencia de resonancia de cada uno de las articulaciones del robot se determina encontrando un control proporcional que permita que el robot intente seguir la señal de entrada (figura B.1), el programa que genera la señal de excitación *woble.m*, cuyo código en Matlab de presenta a continuación.

```

% woble
T0 = 0;
T2 = 2.0;
V = 0.1;
Fmin = 0.5;
Fmax = 30;
Tem=0.001;
Tech = 0.001;
instant=[0.001:Tech:2]';

sortie=[];

Tw = T2 - T0;

for t = 0.001:0.001:2;
if (t - T0)>=0 & (t - T0)<=Tw
f = ((Fmax - Fmin)/Tw)*((t - T0) + Fmin);
res = V*sin(2*pi*f*(t - T0));
end
sortie = [sortie,res];
end

cons1=sortie';
cons2=sortie';
cons3=sortie';
cons4=sortie';

```

Una vez sintonizado este control proporcional el cual se realizó por ensayo y error, se obtuvo el resultado que se muestra en la figura B.2, los picos que se presentan en la señal determinan el momento en el cual el sistema entra en resonancia, y con ayuda del programa que se desarrollo para generar la función de excitación, se determino la frecuencia de resonancia para cada articulación del robot. Así:

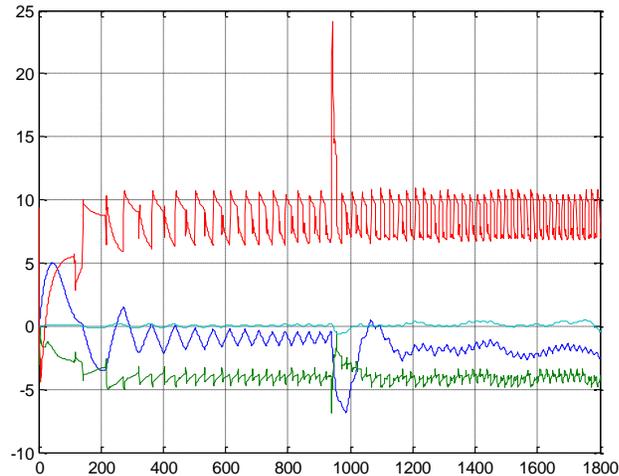


Figura B.2: respuesta del SCARA a una entrada senoidal de frecuencia variable.

Se determina el tiempo en el cual se presenta el pico para cada articulación. La tabla B.1 muestra esta información.

Tabla B.1: Tiempo de detección de sobre picos.

Articulación	Tiempo del inicio de pico
1 (Verde)	938.7 mseg
2 (Cian)	936.6 mseg
3 (Azul)	937.5 mseg
4 (Roja)	943 mseg

Se determina la frecuencia que tiene la señal senoidal de entrada en el instante en el cual se identificó los picos de cada una de las articulaciones, la ecuación (B.1) determina la forma de hallar esta frecuencia.

$$f_r = \left(\frac{f_f - f_i}{T_w} \right) (t + 0.1) \quad (\text{B.1})$$

Donde f_i : frecuencia de inicio de la señal senoidal de entrada (0.1 Hz).
 f_f : frecuencia que alcanza la señal senoidal de entrada (30 Hz).
 t : tiempo en el cual inicia el sobre pico.
 T_w : tiempo de duración de la señal de entrada (2 seg.).

Realizando los cálculos con los datos obtenidos de la figura B.2, se obtiene:

Tabla B.2: Frecuencias de resonancia.

Articulación	Frecuencia de resonancia
1 (Verde)	97.5691 Rad/seg
2 (Fucsia)	97.3718 Rad/seg
3 (Azul)	97.4560 Rad/seg
4 (Roja)	97.9725 Rad/seg

B1.1 Máximos de la diagonal principal

Para encontrar los máximos de la diagonal principal de M , se almacenaron en un arreglo todos los datos de las posiciones del robot durante todo el tiempo de ejecución del programa *Robot_cartesiano_4gl.mdl* (Anexo A); una vez terminada la simulación se ejecutó el programa *Aj_max.m* (Anexo A) el cual calcula el máximo de cada valor de la diagonal principal de M . la tabla B.3 presenta los datos obtenidos.

Tabla B.3: Máximos de la diagonal de M .

Elemento de A_{ij}	Máximos
a_{11}	4.1251
a_{22}	0.2972
a_{33}	0.1
a_{44}	0.045

B.1.2. Sintonización del control PID.

Con la información de la tabla B.2 y B.3 y aplicando las formulas de (3.5) a (3.7) se encuentra las constantes del controlador PID, las cuales se presentan en la tabla B.4.

Tabla B.4: Constantes controlador PID.

Articulación	k_p	k_d	k_i
1	2.9453×10^4	603.7	3.8315×10^6
2	2.1134×10^3	43.4083	3.4386×10^4
3	712.325	14.6184	9.2561×10^4
4	323.95	6.6132	661.2187

En la figuras B.3 y B.4 se presentan los resultados del error de seguimiento del SCARA controlado con el PID, en el caso de la trayectoria circular se presenta el error en estado transitorio y estacionario (figura B.3), para la trayectoria lineal se muestra el error que se presenta en el cambio de dirección (figura B.4).

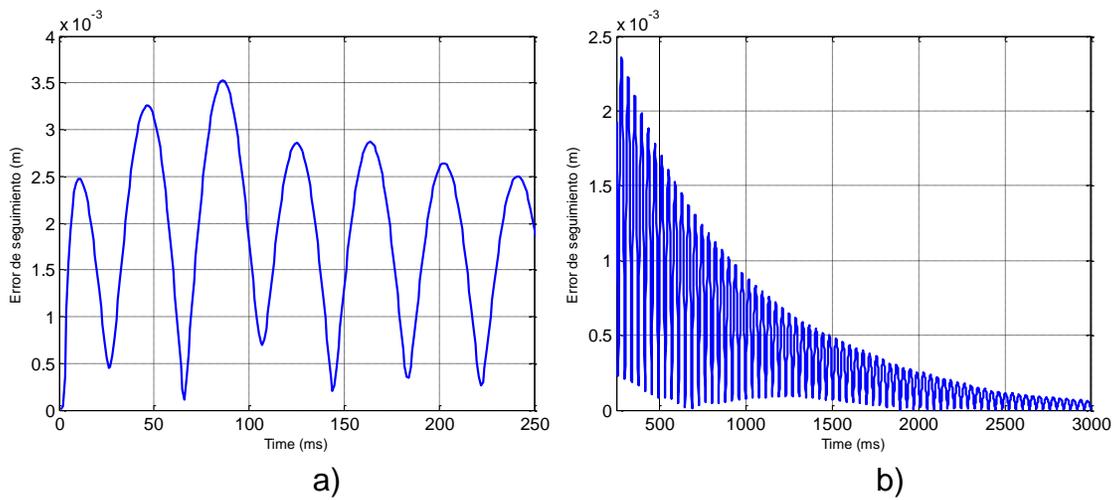


Figura B.3: Error de seguimiento trayectoria circular control PID.
a) Transitorio. b) Estacionario

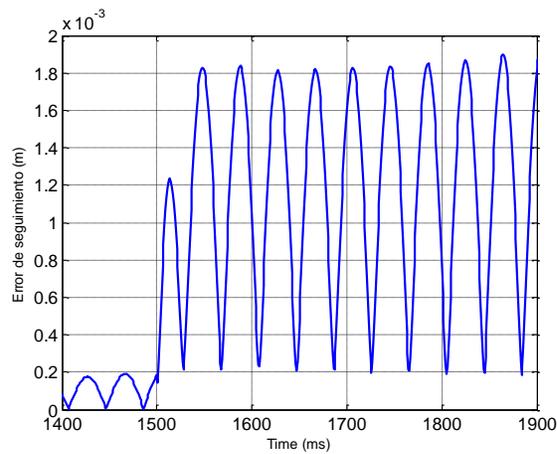


Figura B.4: Error de seguimiento para el cambio de dirección control PID.

B1.3. Sintonización del CTC

Para la sintonización del control CTC se reemplaza (3.12) en (3.11) y se tiene:

$$\begin{aligned}
 \ddot{q} &= \ddot{q}_d + K_v(\dot{q}_d - \dot{q}) + K_p(q_d - q) \\
 (\ddot{q}_d - \ddot{q}) + K_v(\dot{q}_d - \dot{q}) + K_p(q_d - q) &= 0 \\
 \ddot{q} + K_v\dot{q} + K_pq &= 0
 \end{aligned} \tag{B.2}$$

Donde $\ddot{q} = q_d - q$

Las ganancias K_{pj} y K_{vj} son escogidas para imponer al error del eje j la dinámica deseada de amortiguamiento ξ_j (en general igual a 1 para obtener una respuesta sin sobreimpulso) y ω_j es la frecuencia de resonancia, la cual ya se obtuvo y se encuentra en la tabla B.2; luego las constantes del controlador CTC se definen (Khalil, 2002).

$$K_{pj} = w_j^2 \quad (B.3)$$

$$K_{vj} = 2\xi_j w_j \quad (B.4)$$

Donde w_j es la mitad de la frecuencia de resonancia (w_{rj}) de la articulación j . Con los datos de la frecuencia de resonancia y las ecuaciones (B.3) y (B.4) se obtienen las constantes K_{pj} y K_{vj} . En la tabla B.5 se consignan los valores de las constantes del controlador CTC.

Tabla B.5: Constantes del controlador CTC.

Articulación	K_p	K_v
1	2379.9	97.5691
2	2370.3	97.3718
3	2374.4	97.4560
4	2399.7	97.9725

Teniendo en cuenta los datos de las constantes del controlador CTC de la tabla B.5, se realiza la simulación con las mismas características de las trayectorias expresadas anteriormente; los resultados se presentan en las figuras B.5 y B.6.

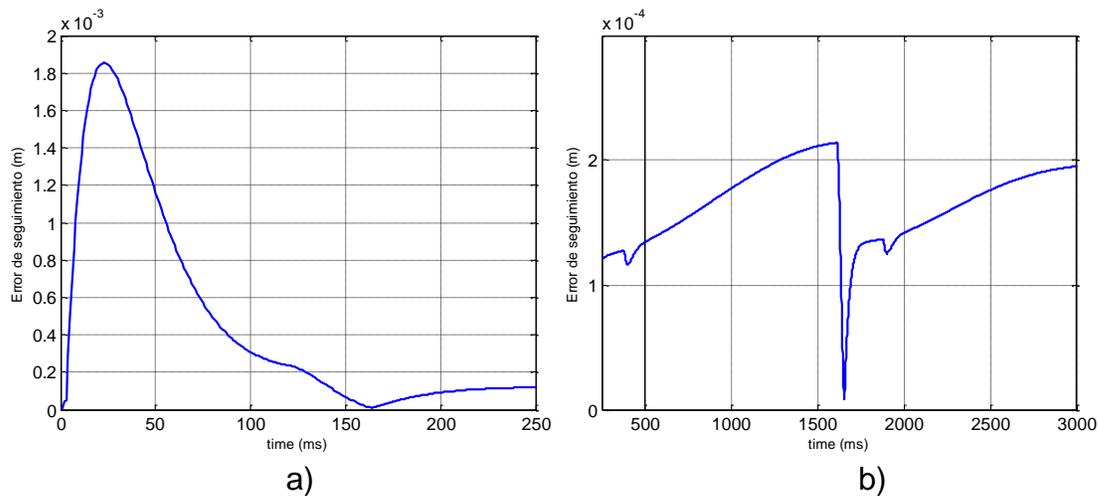


Figura B.5: Error de seguimiento trayectoria circular control CTC. Transitorio. b) Estacionario

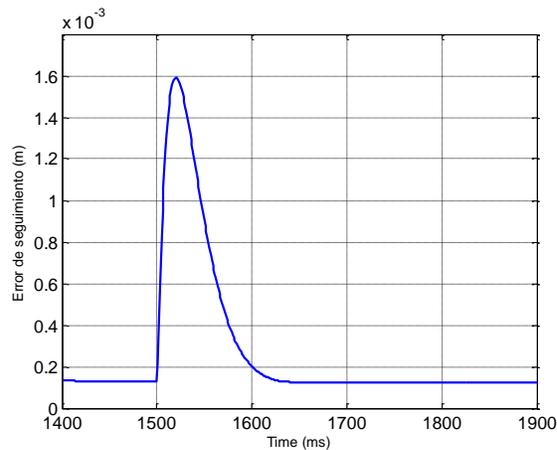


Figura B.6: Error de seguimiento para el cambio de dirección control *CTC*.

B.2. Sintonización mediante el Bloque *Signal Constrain*.

En este aparte se presenta el procedimiento para realizar la sintonización de los controladores con ayuda del bloque de *Simulink Signal Constrain*. Este procedimiento es similar para las técnicas de control de esta tesis.

En la figura B7 se presenta el esquema en bloques de la sintonización del control PID. La diferencia que se presenta en este esquema es la porción del bloque *Signal Constrain*, conectado al error de seguimiento, para la sintonización del controlador en primera instancia se configura este bloque. Definiendo en primera instancia los límites de error en estado transitorio y estacionario que se desean, como se muestra en la figura B.8.

Una vez definido el error de seguimiento deseado, se ejecuta el programa *inicio.m* que se presenta en el anexo A y posteriormente en la pestaña de *optimization*, en la opción de parámetros para sintonizar se adicionan las constantes del controlador. Posteriormente se ejecuta el bloque de *Signal Constrain*, el cual asigna valores a los parámetros seleccionados y verifica que se cumpla con el error de seguimiento deseado, si no es así se asignan nuevos valores al controlador y se ejecuta el programa del control PID, este proceso se lleva a cabo hasta que se encuentran los valores del controlador que cumplan con las especificaciones deseadas del error de seguimiento.

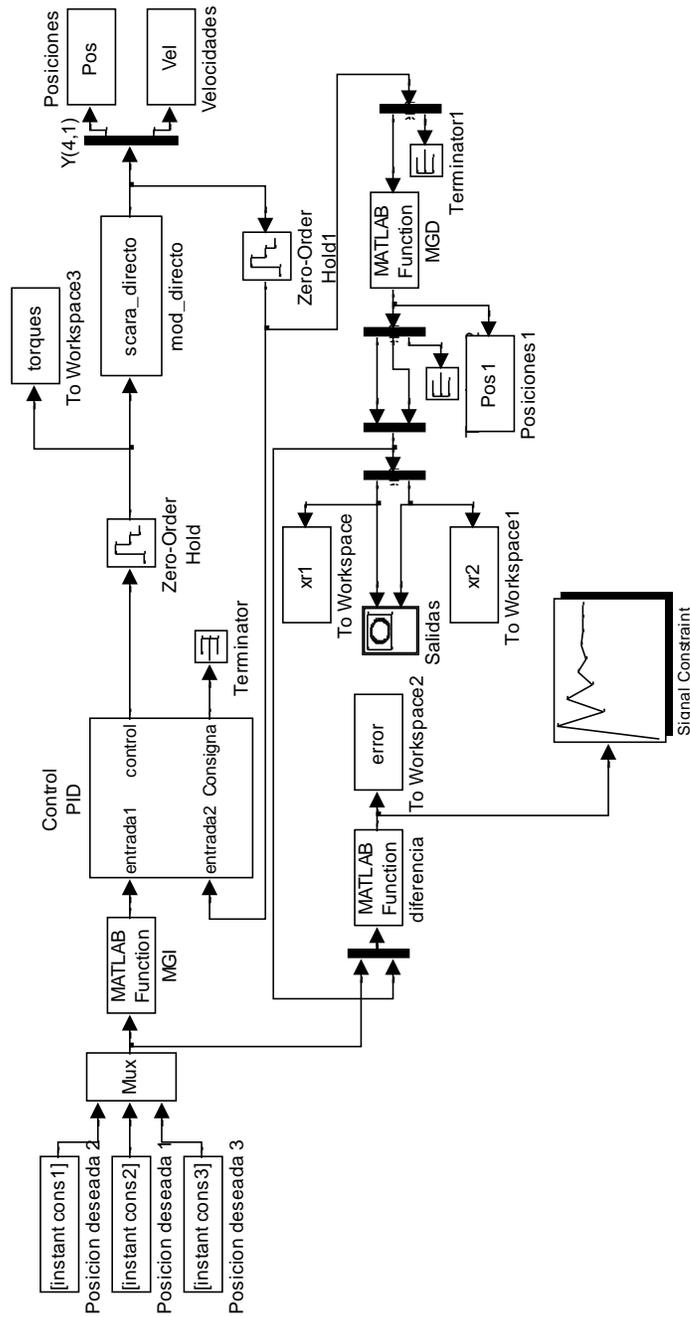


Figura B.7: Esquema de sintonización del control PID.

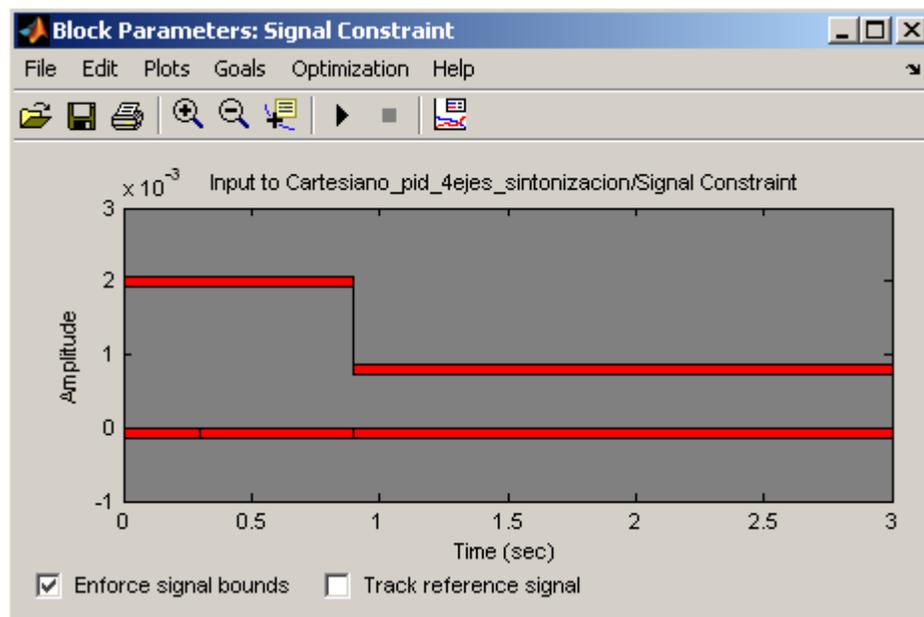


Figura B.8: Configuración de error de seguimiento.