



UNIVERSIDAD VERACRUZANA

Instituto de Ingeniería  
Universidad Veracruzana

Instituto de Ingeniería

“Análisis, diseño y simulación de algoritmos de control, aplicados a un sistema no lineal mecánico (Robot de dos grados de libertad).”



TESIS

QUE PARA OBTENER EL GRADO DE

MAESTRO EN

INGENIERIA ELECTRICA OPCION CONTROL

Presenta

*Ing. Miguel Angel Martínez Vidal*

ASESOR

M.C. Vicente Miguel Díaz de León Santiago

H. VERACRUZ, VER.

ENERO 2001.

# Tesis de Maestría



## UNIVERSIDAD VERACRUZANA INSTITUTO DE INGENIERIA

H. Veracruz, Ver., a 24 de Enero del 2001  
DI021/01

Al Candidato al Grado:  
ING. MIGUEL ANGEL MARTINEZ VIDAL  
PRESENTE:

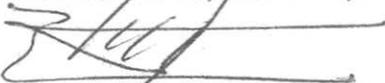
En atención a su solicitud relativa, me es grato transcribir a Usted a continuación el tema que aprobado por esta Dirección propuso el M.C. Vicente Miguel Díaz de León Santiago, para que lo desarrolle como tesis, para obtener el Grado de Maestro en Ingeniería Eléctrica Opción Control:

TEMA:  
"ANÁLISIS, DISEÑO Y SIMULACIÓN DE ALGORITMOS DE CONTROL,  
APLICADOS A UN SISTEMA NO LINEAL MECANICO (ROBOT DE DOS GRADOS  
DE LIBERTAD)."

- I .- Introducción
- II .- Planteamiento del problema
- III .- Marco teórico conceptual
- IV .- Análisis y diseño
- V .- Simulación y resultados
- Conclusión
- Bibliografía
- Apéndice

Sin otro particular, me es grato quedar de Usted como su atento y seguro servidor.

ATENTAMENTE  
"LIS DE VERACRUZ: ARTE, CIENCIA, LUZ"

  
DR. ENRIQUE A. MORALES GONZÁLEZ  
DIRECTOR

EMG/apm\*

Instituto de Ingeniería  
Universidad Veracruzana



**“La experiencia es una maestra  
muy exigente: primero pone la  
prueba y después da la lección”**

*Eric*

## DEDICATORIAS

### A Dios

Por todo lo que me ha dado sin merecerlo

### A mis padres Tereso y Charito

Porque esto es fruto de sus ilusiones, trabajo, pero sobre todo de su amor

### A mis hermanos Eric y Crystian

Por su apoyo y cariño en todos los días de mi vida

### A mis abuelitos:

Ruperto

María (†)

Zenaido

Chica

Por esos consejos tan importantes que me han guiado tanto

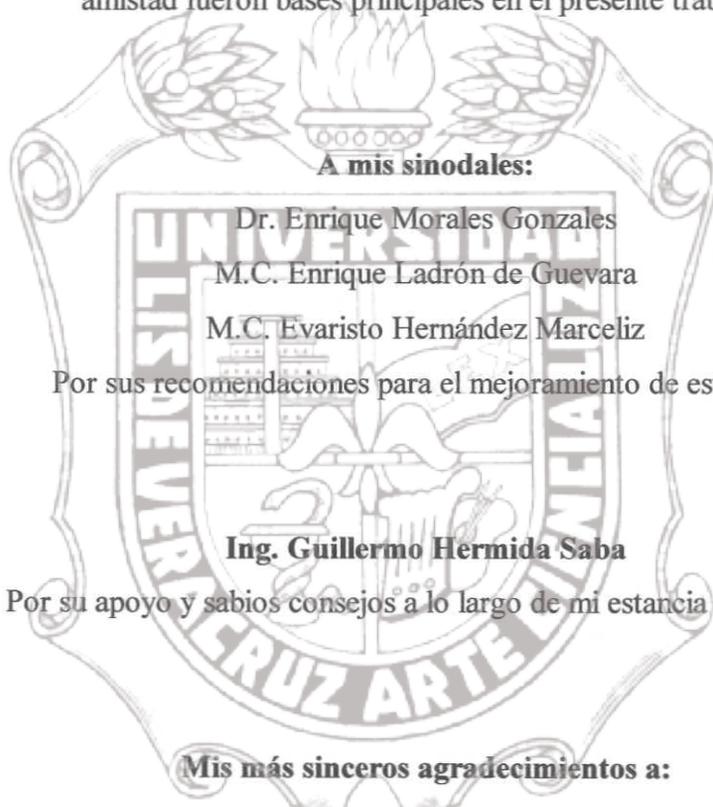
### A mis familiares y amigos

Por esas muestras de cariño y afecto que me dan en todo momento

## RECONOCIMIENTOS

**Al Dr. Vicente Miguel Díaz de León**

Pilar fundamental en el desarrollo de esta tesis, donde sus conocimientos, apoyo y amistad fueron bases principales en el presente trabajo.



**A mis sinodales:**

Dr. Enrique Morales Gonzales

M.C. Enrique Ladrón de Guevara

M.C. Evaristo Hernández Marceliz

Por sus recomendaciones para el mejoramiento de esta tesis

**Ing. Guillermo Hermida Saba**

Por su apoyo y sabios consejos a lo largo de mi estancia académica

**Mis más sinceros agradecimientos a:**

Ing. Genaro Martínez García

Ing. Gerardo Enrique Pascual Torres

Ing. Juan Gabriel Murrieta Cruz

Ing. David Ovando Escobar

Por su gran amistad y apoyo incondicional, que me han dado en mi vida privada y a lo largo de mi trayectoria profesional

## INDICE GENERAL

	Página
<b>RESUMEN</b>	1
<b>ABSTRACT</b>	2
<b>I. INTRODUCCIÓN</b>	3
<b>II. PLANTEAMIENTO DEL PROBLEMA</b>	5
2.1 Objetivos	5
<b>III. MARCO TEÓRICO CONCEPTUAL</b>	6
3.1 Automatización y robótica	6
3.2 Definición y clasificación del robot	9
3.2.1 Definición del robot industrial	10
3.2.2 Clasificación del robot industrial	13
3.2.3 Robots de servicio y teleoperandos	14
3.3 Fundamentos de la tecnología del robot, programación y aplicaciones	15
3.3.1 Anatomía del robot	16
3.3.2 Estructura mecánica de un robot	17
3.3.2.1 Cuatro configuraciones de robots comunes.	17
3.3.2.2 Movimientos del robot.	20
3.4 Introducción a la teoría de control	23
3.4.1 Sistemas lineales en variables de estado	23

3.4.2	Sistemas no lineales en variables de estado	27
3.4.3	Conceptos de estabilidad	30
3.4.3.1	Puntos de equilibrio	30
3.4.3.2	Estabilidad de Lyapunov	31
3.4.3.3	Teorema de LaSalle	33
<b>IV.</b>	<b>ANÁLISIS Y DISEÑO</b>	<b>34</b>
4.1	Análisis	34
4.1.1	Dinámica del brazo de 2 articulaciones (Estructura y propiedades)	35
4.2	Diseño	54
4.2.1	Par calculado	54
4.2.2	Proporcional derivativo (PD)	57
4.2.3	Proporcional-Integral-Derivativo (PID par calculado)	60
4.2.4	Control basado en la Estructura Pasiva de los robots rígidos	61
4.2.4.1	Introducción	61
4.2.4.2	Diseño	62
4.2.5	Robusto adaptable	64
4.2.5.1	Introducción	64
4.2.5.2	Diseño	65
4.2.6	Robusto. Con compensación a la incertidumbre paramétrica, o control tipo "saturación".	68

4.2.6.1	Diseño	68
4.2.6.2	Análisis de estabilidad	70
4.2.7	Control Robusto de Modo Deslizante (1ª Forma)	73
4.2.8	Control Robusto de Modo Deslizante (2ª Forma)	75

<b>V.</b>	<b>SIMULACIÓN Y RESULTADOS</b>	<b>76</b>
5.1	El programa Matlab	76
5.2	Simulación en lazo abierto	79
5.3	Simulación en lazo cerrado	83
5.3.1	Regulación.	84
5.3.1.1	Control PD par-calculado y PD (por Lyapunov).	84
5.3.1.2	Control PID y Estructura pasiva.	86
5.3.1.3	Control Adaptable.	88
5.3.1.4	Control Robusto compensado y de modo Deslizante.	89
5.3.1.5	Deslizante (2ª opción)	91
5.3.2	Seguimiento.	93
5.3.2.1	Control PD par-calculado y PD (por Lyapunov).	93
5.3.2.2	Control PID par-calculado y Estructura Pasiva.	94
5.3.2.3	Adaptable.	96
5.3.2.4	Control Robusto compensado y de modo Deslizante.	97

CONCLUSIÓN	98
BIBLIOGRAFÍA	99
APÉNDICE A	100
APÉNDICE B	112
INDICE DE FIGURAS	123
INDICE DE TABLAS	131



## RESUMEN

Uno de los métodos de automatización en el mundo, es la implementación de diferentes sistemas robóticos en la industria, pero esto no es lo primordial porque cualquiera podría fabricarlo, lo importante es que éste haga lo que se le “ordene”. He ahí donde actúa esta importante rama de la ciencia que es aplicable a diferentes áreas, EL CONTROL, el cual se encarga de diseñar ciertos algoritmos [1] para que un sistema siga una referencia de entrada. En esta tesis se le llama sistema a un robot manipulador de dos grados de libertad, el cual se estudia y modela matemáticamente analizando sus propiedades.

Se diseñan diferentes controladores utilizando el método de la estabilidad de Lyapunov y linealización, garantizando la estabilidad del sistema y por lo tanto el seguimiento de las trayectorias deseadas o referencia de entrada. Se examina el funcionamiento de los diferentes controles diseñados, en un software muy poderoso que simplifica la operación con matrices y escalares (Matlab[2]); se comparan por medio de gráficas para saber cual es el conveniente en ciertos trabajos y se observa como un control PD[1] tan simple como parece, puede tener mejores resultados que otros controles más complicados en su diseño; se ve como algunos controles no se pueden comparar entre sí porque tienen diferentes objetivos de control. Cabe hacer mención que esto no es una novedad, pero que en nuestro país no se fomenta o no se le da la importancia que se merece.

## ABSTRACT

One of the automatization methods in the world is the implementation of different robotic systems in the industry, but this is not the primordial thing because anyone could make it, the important is that it does what is ordered to do. Here is where this important branch of science that is useful in different areas acts, THE CONTROL, which is in charge of designing some algorithms [1] for making a system following an input reference. In this thesis a manipulator robot with 2-degrees-of-freedom is called system, which is mathematically studied and modeled analyzing its features.

Different controllers are designed using Lyapunov stability and Feedback-Linearization method, to guarantee the system's stability and thus the following of the desired trajectories or input reference. The function of the different design controls is examined using powerful software, which simplifies matrices and scalars operations (Matlab [2]). They are compared by graphics to know which is the convenient one in some works and we show how a PD control [1] as simple as it seems, can get better results than other more complicated ones in its design controls; can see how some controls can not be compared between each other because they have different control objectives. We shall mention that this is not new, but in our country, we do not give the importance that it deserves.

## I. INTRODUCCION

Las personas que están en contacto directo con la industria y que por ciertos factores están atrasados tecnológicamente entre el siglo XX y XXI, se han visto en la necesidad de aprender o actualizarse en escasos años de nuevos conceptos innovadores, muy significativos por su alto contenido tecnológico. Tal es el caso de las computadoras (ordenadores), éstas, hoy en día en su versión personal se han introducido en todas las áreas de la ciencia y las personas van conociendo su modo de uso, su evolución y su amplio campo de posibilidades. Muy a pesar de este gran avance científico, la industria requiere de mayor productividad y está necesitando cada vez más de excelente calidad, rapidez, etc. Lo que obliga a la ciencia a romper estas barreras y crear otros conceptos tecnológicos. De donde destaca mencionar el concepto de *ROBOT*.

Esta tesis, ha sido realizada debido a la constante preocupación por la automatización de los diferentes procesos, en el amplio ámbito de la industria; como ingeniero en electrónica veo con gran tristeza pero con mucha esperanza, que nuestro país es una gran potencia mundial en cuanto a tecnología se refiere, pero como tal tiene el gran defecto de obtenerla muchos años después. Posiblemente sea el miedo a invertir en el campo del conocimiento y del saber, o quizás sea la "dejadez". En fin, el caso es que si se desea que los productos generados por la industria sean competitivos a un nivel mundial, se debe estar actualizados tecnológicamente, es decir, aprender a automatizar los procesos y tener un control adecuado acorde a las necesidades. Debido al poco desarrollo existente en la robótica en México, se hace necesario e importante que se realicen investigaciones en esta área tan interesante, ya que la robótica contiene grandes capacidades de aplicaciones sobre la tecnología industrial.

# Tesis de Maestría

La robótica genera a las empresas una mayor productividad y seguridad, esto es claro de observar, si se analizan los avances que tienen los países desarrollados; esto sucede por ser la robótica una de las tecnologías que más satisfactorios ha producido para el ser humano, ya sea en forma de diversiones o como productos domésticos, el robot se utiliza en vuelos espaciales o en industrias de alto riesgo humano, así como en sistemas de comunicaciones, en producción de energía, en la industria automotriz, etc. En esta tesis, el enfoque es más sobre la parte del control, donde se hacen diversas pruebas con diferentes tipos de controles como PD, PID, adaptable, deslizante.



## II. PLANTEAMIENTO DEL PROBLEMA.

En la actualidad, existe más de un tipo de controladores para sistemas en las diversas áreas de la tecnología moderna, ya sea mecánicos, electrónicos, biomédicos, etc. Conforme pasa el tiempo y la ciencia avanza, se están implementando nuevas técnicas de control, unas más complejas otras más sencillas pero también más deficientes o no recomendables para una determinada planta. En este trabajo de investigación necesario saber ¿qué controlador es el más adecuado para tener el mejor control de un sistema?, el cual está muy involucrado por las áreas mecánica, eléctrica, electrónica, sistemas computacionales y algunas otras. Por consiguiente, es posible obtener un comparativo de las técnicas de control para un sistema no lineal (Robot manipulador de 2 grados de libertad); y poder observar como funcionan algunas técnicas de control lineal como: PD, PID, con respecto a otras técnicas de diseño más complejas[3], las cuales consideran la presencia de parámetros no conocidos y de perturbaciones externas. Por otro lado y como complemento de este trabajo más adelante se puede desarrollar un prototipo e implementar alguno de estos algoritmos en tiempo real.

### 2.1 OBJETIVOS.

- Analizar, diseñar y Simular algunos algoritmos de control conocidos como: PD, PID, Adaptable, Deslizante; aplicados a un robot manipulador de 2 grados de libertad.
- Comparar el desempeño de cada uno de estos controladores en simulación.

## III. MARCO TEORICO CONCEPTUAL

### 3.1 Automatización y robótica.

La automatización y la robótica son dos tecnologías muy ligadas. En un contexto industrial se puede definir la automatización como: “una tecnología que está relacionada con el empleo de sistemas mecánicos, electrónicos y basados en computadoras para la operación y control de la producción”. Algunos ejemplos de esta tecnología son: líneas de transferencias, máquinas de montaje mecanizado, sistemas de control de realimentación (aplicados a los procesos industriales). Basándose en esta definición, la robótica es una forma de automatización industrial.

Hay tres formas clásicas de automatización industrial: *automatización fija*, *automatización programable* y *automatización flexible*[4]. La *automatización fija* se utiliza cuando el volumen de producción es muy alto, por lo tanto, es adecuada para diseñar equipos especializados para procesar el producto (o componentes de un producto) con alto rendimiento o con elevadas tasas de producción. Un buen ejemplo de la automatización fija puede encontrarse en la industria del automóvil, en donde líneas de transferencia muy integradas constituidas por varias decenas de estaciones de trabajo se utilizan para operaciones de mecanizado en componentes y transmisiones. La economía de la automatización fija es tal, que el coste de los equipos especiales pueden dividirse entre un gran número de unidades y los costes unitarios resultantes son bajos en relación con los métodos de producción alternativos. Un problema con la automatización fija es que el equipo está especialmente diseñado para obtener el producto, y una vez que se haya acabado el ciclo de vida del producto es probable que el equipo quede obsoleto. Para productos con corto ciclo de vida el empleo de la automatización fija representa un gran riesgo.

La *automatización programable* se emplea cuando el volumen de producción es relativamente bajo y hay una gran diversidad de producción a obtener. En este caso, el equipo de producción está diseñado para ser adaptable a variaciones en la configuración del producto. Esta característica se realiza haciendo funcionar el equipo bajo el control de un <<programa>> de instrucciones que se preparó especialmente para el producto dado. El programa se introduce por lectura en el equipo de producción y este último realiza la secuencia particular de operaciones de procesamiento(o montaje) para obtener el producto.

Existe una categoría entre la automatización fija y la programable que se denomina *automatización flexible*. Otros términos utilizados para la automatización flexible incluyen los 'sistemas de fabricación flexibles' (o FMS) y los 'sistemas de fabricación integrados por computadora'. El concepto de automatización flexible solo se desarrolló en la práctica en los últimos quince o veinte años. La experiencia adquirida hasta ahora con este tipo de automatización indica que es más adecuado para el rango de producción de volumen medio. Los sistemas flexibles tienen algunas características de la automatización fija y de la automatización programable. Debe programarse para diferentes configuraciones de productos, pero la diversidad de las configuraciones suele estar más limitada que para la automatización programable, lo que permite que se produzca un cierto grado de integración en el sistema. Los sistemas automatizados flexibles suelen estar constituidos por una serie de estaciones de trabajo que están interconectadas por un sistema de almacenamiento y manipulación de materiales. Una computadora central se utiliza para controlar las diversas actividades que se producen en el sistema, encaminando las diversas piezas a las estaciones adecuadas y controlando las operaciones programadas en las diferentes estaciones.

Una de las características que distingue a la automatización programable de la flexible es que con la automatización programable los productos se obtienen en lotes. Cuando se completa un lote, el equipo se reprograma para procesar el siguiente lote. Con la automatización flexible, diferentes productos pueden obtenerse al mismo tiempo en el mismo sistema de fabricación. Esta característica permite un nivel de versatilidad que no está disponible en la automatización programable pura, como se definió anteriormente. Esto significa que pueden obtenerse productos en un sistema flexible en lotes si ello fuera deseable, o varios estilos de productos diferentes pueden mezclarse en el sistema. La potencia de cálculo de la computadora de control es lo que posibilita esta versatilidad.

De los tres tipos de automatización, la robótica coincide más estrechamente con la automatización programable. Un *robot industrial* es una máquina programable de uso general que tiene algunas características antropomorfas o 'humanoides'. La característica humanoide más típica de los robots actuales es la de sus brazos móviles. El robot puede programarse para desplazar su brazo a través de una secuencia de movimientos con el fin de realizar alguna tarea de utilidad. Repetirá este modelo de movimientos una y otra vez hasta que se re programe para ejecutar alguna otra tarea. Por consiguiente, la característica de programación permite que los robots se utilicen para una diversidad de operaciones industriales diferentes, muchas de las cuales implican el trabajo del robot junto con otros elementos de equipos automatizados o semiautomatizados. Estas operaciones incluyen carga y descarga de máquina, la soldadura por puntos y la pintura por pulverización.

La definición oficial de un robot industrial se proporciona por la Robotics Industries Association (RIA), anteriormente el Robotics Institute of American (RIA)[4]:

“Un robot industrial es un manipulador multifuncional reprogramable diseñado para desplazar materiales, piezas, herramientas o dispositivos especiales mediante movimientos programados variables para la ejecución de una diversidad de tareas”.

Esta definición refuerza la conclusión de que los robots industriales deben clasificarse como una forma de automatización programable. Los programas de los robots están contenidos en la computadora o controlador programable y se cargan en cada robot para la carrocería de automóvil particular que ha de soldarse en cada estación de trabajo.

Actualmente la analogía humana de un robot industrial es muy limitada. Los robots no tienen apariencia humana y no se comportan como seres humanos. Se trata más bien de máquinas de un solo brazo que casi siempre operan desde una posición fija en la superficie de la fábrica. Es probable que los futuros robots tengan un mayor número de atributos, similares a los humanos. También es probable que tengan mayores capacidades de sensores, más inteligencia, un nivel más alto de destreza manual y un grado limitado de movilidad.

### ***3.2 Definición y clasificación del robot.***

La robótica está experimentando en la actualidad una notoria ampliación en sus campos de actuación. Saturadas en cierta medida las aplicaciones de los robots en talleres y líneas de fabricación, los robots están tomando posiciones en aplicaciones fuera

del taller, caracterizadas fundamentalmente por la falta de estandarización del producto y del proceso y la no-estructuración del entorno.

Buena parte de las definiciones y clasificaciones de los robots existentes responden al robot ampliamente utilizado hasta la fecha, destinado a la fabricación flexible de series medias y que se conoce como *robot industrial* o robot de producción. Los robots especiales también denominados robots de servicio, están aún en un estado de desarrollo incipiente. Las definiciones y generaciones aquí presentadas responden fundamentalmente al robot industrial.

### 3.2.1 Definición del robot industrial.

Existen ciertas dificultades a la hora de establecer una definición formal de lo que es un robot industrial. La primera de ellas surge de la diferencia conceptual entre el mercado japonés y el euro-americano de lo que es un robot y un manipulador. Así, mientras que para los japoneses un robot industrial es cualquier dispositivo mecánico dotado de articulaciones móviles destinado a la manipulación, el mercado occidental es más restrictivo, exigiendo una mayor complejidad, sobre todo en lo relativo al control.

En segundo lugar, y centrándose ya en el concepto occidental, aunque existe una idea común acerca de lo que es un robot industrial, no es fácil ponerse de acuerdo a la hora de establecer una definición formal. Además, la evolución de la robótica ha ido obligando a diferentes actualizaciones de su definición.

La definición más comúnmente aceptada posiblemente sea la de la *Asociación de Industrias Robóticas (RIA)*[5], según la cual:

- “Un robot industrial es un manipulador multifuncional reprogramable, capaz de mover materias, piezas, herramientas o dispositivos especiales, según trayectorias variables, programadas para realizar tareas diversas”.

Esta definición, ligeramente modificada, ha sido adoptada por la organización internacional de estándares (ISO) que define al robot industrial como:

- “Manipulador multifuncional reprogramable con varios grados de libertad, capaz de manipular materias, piezas, herramientas o dispositivos especiales según trayectorias variables programadas para realizar tareas diversas”.

Se incluye en esta definición la necesidad de que el robot tenga varios grados de libertad. Una definición más completa es la establecida por la *Asociación Francesa de Normalización* (AFNOR)[5] que define primero el manipulador y, basándose en dicha definición, el robot:

- Manipulador: “mecanismo formado generalmente por elementos en serie, articulados entre sí, destinado al agarre y desplazamiento de objetos. Es multifuncional y puede ser gobernado directamente por un operador humano o mediante dispositivo lógico”.
- Robot: “manipulador automático servocontrolado, reprogramable, polivalente, capaz de posicionar y orientar piezas, útiles o dispositivos especiales, siguiendo trayectorias variables reprogramables, para la ejecución de tareas variadas. Normalmente

tiene la forma de uno o varios brazos terminados en una muñeca. Su unidad de control incluye un dispositivo de memoria y ocasionalmente de percepción del entorno. Normalmente su uso es el de realizar una tarea de manera cíclica, pudiéndose adaptar a otra sin cambios permanentes en su material”.

Por último, la *Federación Internacional de Robótica* (IFR) en su informe técnico ISO/TR 83737 (septiembre 1988) [5] distingue entre robot industrial de manipulación y otros robots:

- “Por robot industrial de manipulación se entiende a una máquina de manipulación automática, reprogramable y multifuncional con tres o más ejes que pueden posicionar y orientar materias, piezas, herramientas o dispositivos especiales para la ejecución de trabajos diversos en las diferentes etapas de la producción industrial, ya sea en una posición fija o en movimiento”

**Tabla 3.1 Clasificación de los robots según la AFRI.**

Tipo A	Manipulador con control manual o telemando.
Tipo B	Manipulador automático con ciclos preajustados; regulación mediante fines de carrera o topes; control por PLC; accionamiento neumático, eléctrico o hidráulico.
Tipo C	Robot programable con trayectoria continua o punto a punto. Carece de conocimientos sobre su entorno.
Tipo D	Robot capaz de adquirir datos de su entorno, readaptando su tarea en función de éstos.

1. Configuración polar.
2. Configuración cilíndrica.
3. Configuración de coordenadas cartesianas.
4. Configuración de brazo articulado.

La configuración polar utiliza un brazo telescópico que puede elevarse o bajar alrededor de un pivote horizontal. Este pivote está montado sobre una base giratoria. Estas diversas articulaciones proporcionan al robot la capacidad para desplazar su brazo dentro de un espacio esférico, y de aquí la denominación de robot de <<coordenadas esféricas>> que se puede aplicar a este tipo. Varios robots comerciales tienen la configuración polar. Entre estos se puede citar al conocido *Unimate* serie 2000, el *MAKER 110*, fabricado por United States Robots.

La configuración cilíndrica utiliza una columna vertical y un dispositivo de deslizamiento que puede moverse hacia arriba o hacia abajo a lo largo de la columna. El brazo del robot está unido al dispositivo deslizante de modo que puede moverse en sentido radial con respecto a la columna. Haciendo girar la columna, el robot es capaz de conseguir un espacio de trabajo que se aproxima a un cilindro. El robot de coordenadas cartesianas, utiliza tres dispositivos deslizantes perpendiculares para construir los ejes  $x$ ,  $y$  y  $z$ . Otros nombres se aplican, a veces, a esta configuración, incluyendo las denominaciones de robot  $xyz$  y robot rectilíneo. Desplazando los tres dispositivos deslizantes entre sí, el robot es capaz de operar dentro de una envolvente rectangular de trabajo. Un ejemplo de esta configuración es el robot IBM RS-1 (actualmente denominado modelo 7565). La configuración del RS-1, debido a su apariencia y construcción, se denomina ocasionalmente de <<caja>>. El robot de <<pórtico>> es otro nombre utilizado para los robots cartesianos, que suelen ser grandes y que tienen la

apariencia de una grúa del tipo de pórtico. El robot de brazo articulado tiene la configuración similar al brazo humano. Está constituido por dos componentes rectos, que corresponden al antebrazo y al brazo humanos, montados sobre un pedestal vertical. Estos componentes están conectados por dos articulaciones giratorias que corresponden al hombro y al codo. Una muñeca está unida al extremo del antebrazo, con lo que se le proporcionan varias articulaciones suplementarias. Varios robots disponibles en el comercio tienen la configuración de brazo articulado, incluyendo al robot Cincinnati Milacron T3 (Modelo 776). Una versión especial del robot de brazo articulado es el SCARA (Selective Compliance Assembly Robot Arm), cuyas articulaciones de hombro y de codo giran alrededor de ejes verticales. SCARA es una configuración que proporciona una importante rigidez para el robot en la dirección vertical, pero una elasticidad en el plano horizontal. Esto le hace ideal para muchas tareas de montaje.

Existen ventajas e inconvenientes relativos a las cuatro anatomías de robot básicas, simplemente debido a sus geometrías. Desde el punto de vista de repetibilidad de movimiento (la capacidad para desplazarse a un punto determinado del espacio con un error mínimo), el robot cartesiano de estructura de caja es probable que tenga ventaja, debido a su estructura inherente rígida. Desde el punto de vista de alcance (la capacidad del robot para extender su brazo significativamente más allá de su base), las configuraciones: polar y de brazo articulado resultan ventajosas. La capacidad de elevación del robot es importante en muchas aplicaciones. La configuración cilíndrica y el robot xyz de pórtico pueden diseñarse para una alta rigidez y gran capacidad del transporte de carga. Para las aplicaciones de carga de máquinas, la capacidad del robot para penetrar a través de una pequeña abertura, sin interferencia con los lados de dicha abertura, es importante. Las configuraciones polar y cilíndrica tienen una ventaja geométrica natural desde el punto de vista de esta capacidad.

## 3.3.2.2 Movimientos del robot[5].

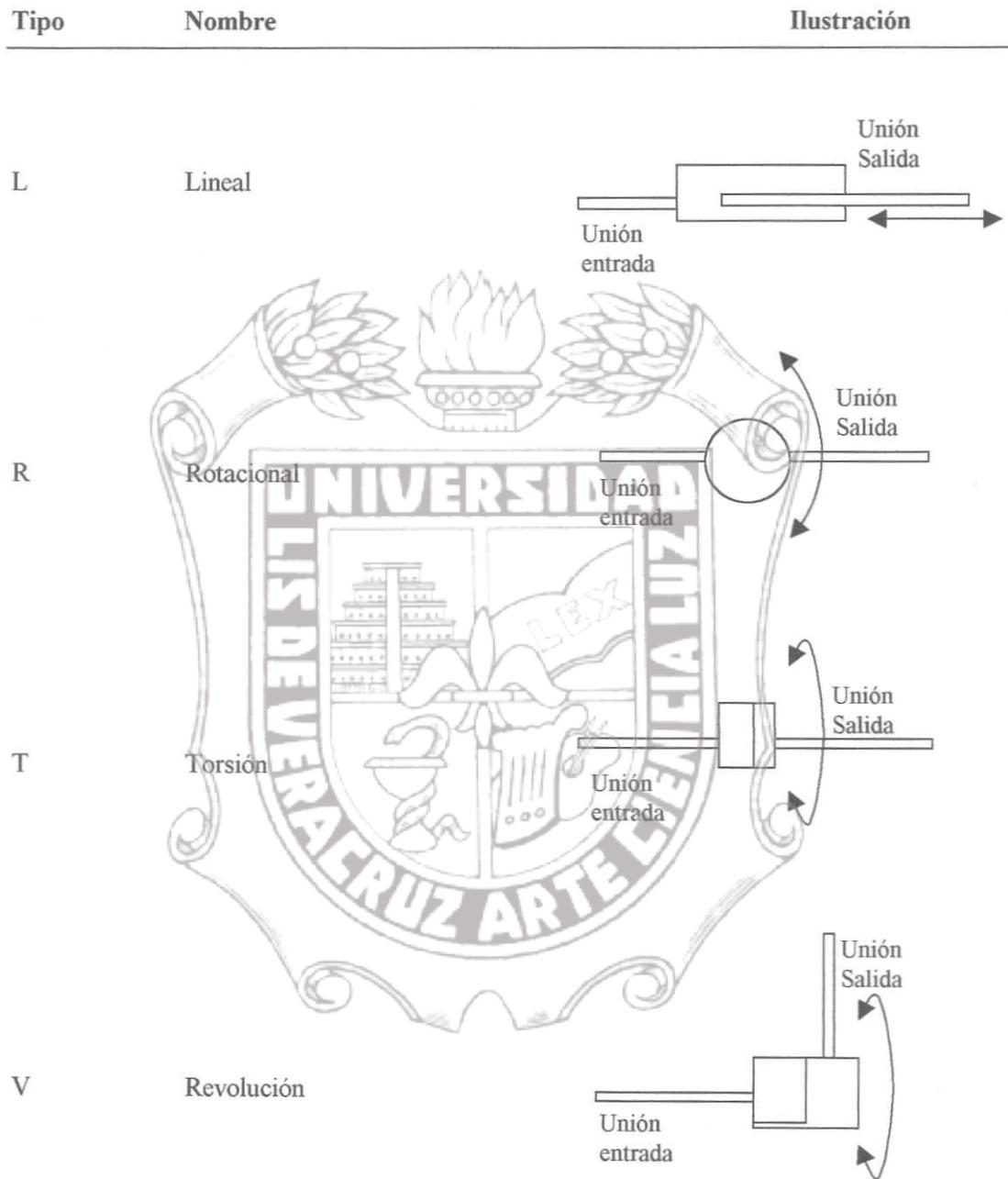
Los robots industriales están diseñados para realizar un trabajo productivo. El trabajo se realiza permitiendo que el robot desplace su cuerpo, brazo y muñeca mediante una serie de movimientos y posiciones. Unido a la muñeca está el efector final, que se utiliza por el robot para efectuar una tarea específica. Los movimientos del robot pueden dividirse en dos categorías generales: movimientos de brazo y cuerpo y movimientos de la muñeca. Los movimientos de articulaciones individuales asociados con estas dos categorías se denominan, a veces, por el término *grado de libertad* (GDL), y un robot industrial típico está dotado de cuatro a seis GDL. Los movimientos de robots se realizan por medio de articulaciones accionadas. Tres articulaciones suelen estar asociadas con la acción del brazo y del cuerpo y dos o tres articulaciones se suelen emplear para accionar la muñeca. Para la conexión de las diversas articulaciones del manipulador se emplean unos elementos rígidos denominados uniones. En cualquier cadena de unión, se le llama unión de entrada, al eslabón que está más próximo a la base en la cadena. La unión de salida es la que se desplaza con respecto a la entrada. Las articulaciones utilizadas en el diseño de robots industriales suelen implicar un movimiento relativo de las uniones contiguas, movimiento que es lineal o rotacional. Las articulaciones lineales implican un movimiento deslizante o de traslación de las uniones de conexión. Este movimiento puede conseguirse de varias formas (por ejemplo, mediante un pistón, un mecanismo telescópico y el movimiento relativo a lo largo de un carril o vía lineal).

La articulación lineal como se muestra en la figura 3.1, se refiere a una articulación de tipo L (como abreviatura de lineal). El término de articulación prismática se utiliza, a veces, en la documentación sobre robótica en lugar de articulación lineal. Hay, como mínimo, tres tipos de articulación giratoria que pueden distinguirse en los manipuladores de los robots: rotacional, de torsión y de revolución. Los tres tipos se

ilustran en la figura 3.1. En la articulación de tipo R (Rotacional) el eje de rotación es perpendicular a los ejes de las dos uniones. El segundo tipo de articulación giratoria, implica un movimiento de torsión entre las uniones de entrada y salida. El eje de rotación de la articulación de torsión es paralelo a los ejes de ambas uniones, a este tipo de articulación se le llama tipo T (Torsión). El tercer tipo de articulación giratoria es un tipo de articulación de revolución en la que la unión de entrada es paralela al eje de rotación y la de salida es perpendicular a dicho eje. Esencialmente, la unión de salida gira alrededor de la entrada como si estuviera en órbita. Esta articulación se designará como de tipo V (revolución).

Las articulaciones del brazo y del cuerpo están diseñadas para permitir al robot desplazar su efector final a una posición deseada dentro de los límites del tamaño del robot y de los movimientos de las articulaciones. Para robots de configuración polar, cilíndrica o de brazo articulado, los tres grados de libertad asociados con los movimientos del brazo y del cuerpo son:

1. Transversal vertical: es la capacidad para desplazar la muñeca hacia arriba o abajo para proporcionar la postura vertical deseada.
2. Transversal radial: implica la extensión o retracción (movimiento hacia dentro o afuera) del brazo desde el centro vertical del robot.
3. Transversal rotacional: es la rotación del brazo alrededor del eje vertical.



**Figura 3.1.** Varios tipos de articulaciones utilizadas en robots, por ejemplo: articulación rotacional con giro a lo largo de un eje perpendicular a los ejes de los miembros del brazo, articulación rotacional con una acción de torsión, articulación de movimiento lineal, que suele conseguirse mediante una acción de deslizamiento, etc.

## 3.4 Introducción a la teoría de control.

El control de manipuladores robóticos es un área madura, todavía muy fructífera en el campo de la investigación, desarrollo y manufactura. Los robots industriales son básicamente dispositivos de agarre y posición. También, un robot es útil para controlar algunos movimientos y fuerzas en el medio ambiente.

Para el control del robot se requieren de los conocimientos de un modelo matemático y de algún tipo de inteligencia para actuar sobre el modelo. El modelo matemático de un robot es obtenido desde las leyes físicas básicas que gobiernan sus movimientos. Más adelante se observan algunos conceptos de la teoría de control que son necesarios para el análisis del modelo.

### 3.4.1 Sistemas lineales en Variables de Estado

Muchos sistemas físicos como son los robots pueden ser descritos por *ecuaciones diferenciales*. Estas ecuaciones, las cuales son obtenidas por leyes físicas fundamentales, suministran el punto de partida para el análisis y control de sistemas.

Un sistema continuo en el tiempo se dice que es lineal si cumple con el principio de superposición; esto es, si la salida  $y_1(t)$  resulta de la entrada  $u_1(t)$  y la salida  $y_2(t)$  resulta de la entrada  $u_2(t)$ , entonces la salida resultante de  $a_1u_1(t) + a_2u_2(t)$  está dada por  $y(t) = a_1y_1(t) + a_2y_2(t)$ , donde  $a_1$  y  $a_2$  son constantes escalares. Los sistemas lineales, SISO (single-input / single-output), continuos e invariante en el tiempo son descritos por la siguiente suma de ecuaciones diferenciales ordinarias de coeficientes constantes [7]:

$$\begin{aligned} & \frac{d^n y(t)}{dt^n} + a_{n-1} \frac{d^{n-1} y(t)}{dt^{n-1}} + \dots + a_1 \frac{dy(t)}{dt} + a_0 y(t) \\ & = b_n \frac{d^n u(t)}{dt^n} + b_{n-1} \frac{d^{n-1} u(t)}{dt^{n-1}} + \dots + b_1 \frac{du(t)}{dt} + b_0 u(t) \end{aligned} \quad (3.4.1-1)$$

Donde  $a_i$  y  $b_i$  son constantes escalares,  $y(t)$  es la salida escalar, y  $u(t)$  es la entrada escalar. Nótese que la entrada  $u(t)$  es diferenciada tanto como la salida  $y(t)$ . Por otro lado, se dice que el sistema es *no-dinámico*. El estado de un sistema es definido como un grupo de variables. Por ley, entonces, el vector de estados tiene todas las variables necesarias para determinar el comportamiento de alguna variable en el sistema.

Por definición, cada vector de estado  $x$  no es único, más adelante aplicaremos esto. Por tanto, si  $x$  es el vector de estados entonces  $\bar{x} = Tx$  donde  $T$  es una matriz invertible de  $n \times n$ . Para el sistema continuo descrito en (3.4.1-1), elegimos el siguiente vector de estados posible:

$$\begin{aligned} \dot{x}_1 &= \dot{x}_1 \\ \dot{x}_2 &= \dot{x}_2 \\ &\vdots \\ \dot{x}_{n-1} &= \dot{x}_{n-1} \\ \dot{x}_n &= -a_0 x_1(t) - a_1 x_2(t) - \dots - a_{n-1} x_n(t) + u(t) \end{aligned} \quad (3.4.1-2)$$

donde  $\dot{x}_i = \frac{dx_i}{dt_i}, i = 1, 2, \dots, n$ . La ecuación de entrada salida se reduce a:

$$y(t) = b_0 x_1(t) + b_1 x_2(t) + \dots + b_{n-1} x_n(t) + b_n u(t) \quad (3.4.1-3)$$

Una forma más compacta de (3.4.1-2) y (3.4.1-3) está dada por:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + bu(t) \\ y(t) &= cx(t) + du(t) \end{aligned} \tag{3.4.1-4}$$

$$A = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ 0 & 0 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ -a_0 & -a_1 & -a_2 & \dots & -a_{n-2} & -a_{n-1} \end{bmatrix} \quad b = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \tag{3.4.1-5}$$

donde

$$c = [b_0 \quad b_1 \quad b_2 \quad \dots \quad b_{n-1}] \quad d = b_n$$

Esta es la representación en variables de estado conocida como *forma canónica controlable*. En general, un sistema lineal, continuo e invariante en el tiempo tendrá más de una entrada y una salida. Donde  $u(t)$  es un vector de  $m \times 1$  y  $y(t)$  es un vector de  $p \times 1$  dimensiones.

Las ecuaciones diferenciales que relacionan a  $u(t)$  y a  $y(t)$  no serán presentadas aquí, pero la representación en variables de estado de un sistema MIMO (multi-input / multi-output) será:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{aligned} \tag{3.4.1-6}$$

Donde  $A$  es de  $n \times n$ ,  $B$  es de  $n \times m$ ,  $C$  es  $p \times n$ , y  $D$  es  $p \times m$ . La forma específica de  $A$ ,  $B$ ,  $C$ ,  $D$  la podemos ver mejor en [5]. El diagrama bloques de (3.4.1-6) es mostrado en la figura 3.2-a. Nótese que, el número mínimo de estados es igual al número requerido de condiciones iniciales en orden para encontrar una solución única de las ecuaciones diferenciales.

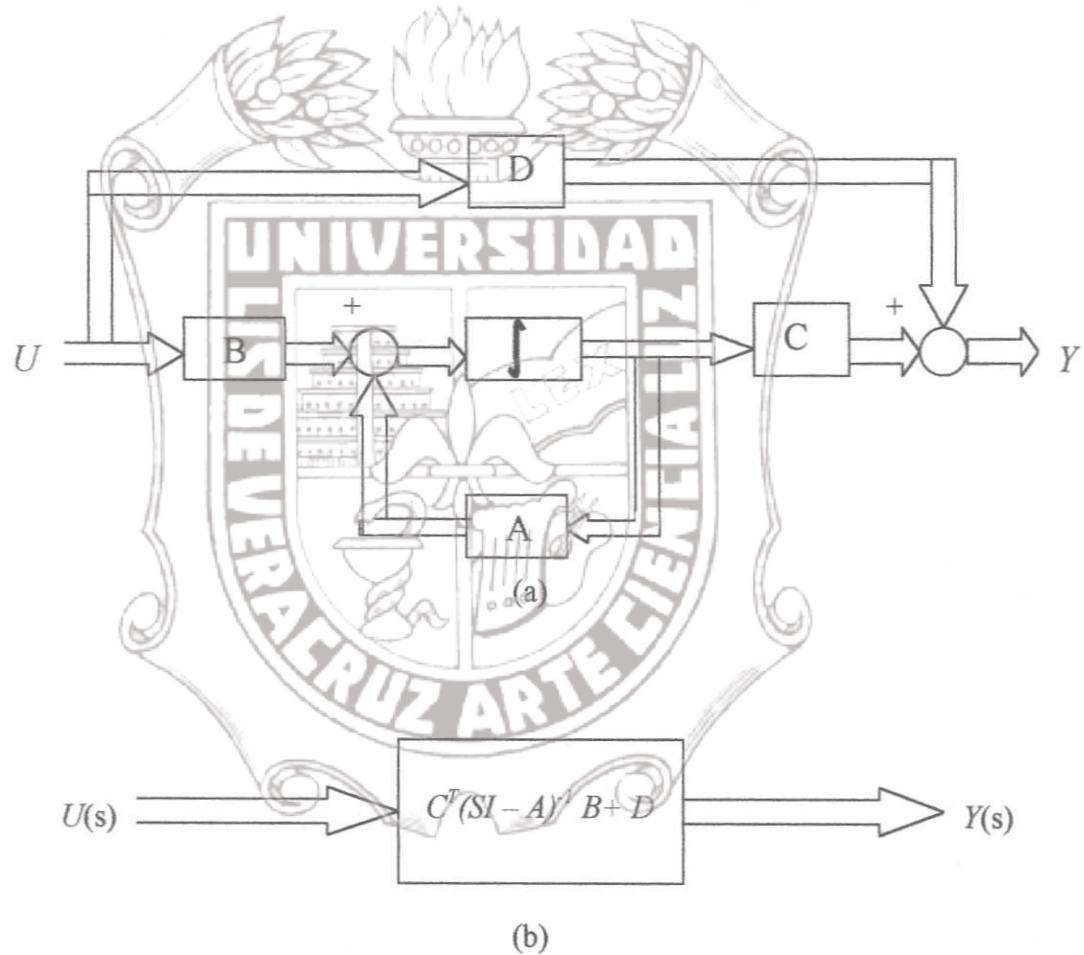


Figura 3.2 (a) Diagrama a bloques del espacio de estado. (b) Diagrama a bloques de (3.4.1-4).

## 3.4.2 Sistemas no lineales en Variables de Estado.

En muchos casos la física fundamental del medio ambiente, no puede ser descrita usando ecuaciones lineales en *variables de estado*. Este es el caso de los robots manipuladores, donde la interacción entre las diferentes articulaciones está descrita por ecuaciones diferenciales no lineales, como se muestra más adelante. La representación en variables de estado es tranquilamente capaz de manejar estos sistemas, mientras que la función de transferencia y métodos en el dominio de la frecuencia fallan.

Un sistema no lineal, escalar, continuo e invariante en el tiempo es descrito por una ecuación diferencial no lineal, escalar y de coeficientes constantes [7] como:

$$\frac{d^n y(t)}{dt^n} = h[y(t), y^{(1)}(t), \dots, y^{(n-1)}(t), u(t), u^{(1)}(t), \dots, u^{(n)}(t)] \quad (3.4.2-1)$$

donde  $y(t)$  es la salida y  $u(t)$  es la entrada al sistema bajo ciertas consideraciones. Como en el caso lineal, se define el vector de estados  $x$  por sus componentes  $x_i$  como sigue:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_3 \\ &\vdots \\ &\vdots \\ \dot{x}_{n-1} &= x_n \\ \dot{x}_n &= h[x_1(t), x_2(t), \dots, x_n(t), u(t), u^{(1)}(t), \dots, u^{(n)}(t)] \end{aligned} \quad (3.4.2-2)$$

La ecuación de salida se reduce a:

$$y(t) = x_1(t) \quad (3.4.2-3)$$

Una forma más compacta de (3.4.2-2) y (3.4.2-3) está dada por:

$$\begin{aligned} \dot{x}(t) &= f[x(t), U(t)] \\ y(t) &= cx(t) \end{aligned} \quad (3.4.2-4)$$

donde

$$U(t) = [u(t) \quad u^{(1)}(t) \quad \dots \quad u^{(n-1)}(t)]^T$$

y

$$c = [1 \ 0 \ 0 \ \dots \ 0]$$

### EJEMPLO 3.4.2-1: Algunos sistemas no lineales

(a) Considere la ecuación de un péndulo

$$\ddot{y} + ky + \sin y = 0$$

Una descripción en variables de estado eligiendo  $x_1 = y, x_2 = \dot{y}$ ,

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ -\sin x_1 - kx_2 \end{bmatrix}$$

En general, los sistemas no lineales tienen  $m$  entradas y  $p$  salidas y hay  $n$  ecuaciones diferenciales descritas por (3.4.2-4), donde  $u$  es un vector de  $m \times 1$ ,  $y$  es un vector de  $p \times 1$ , y  $c$  es una matriz de  $p \times n$ .

Instituto de Ingeniería  
Universidad Veracruzana

## EXEMPLO 3.4.2-2: Ecuaciones dinámicas de Robots Rígidos.

Un robot rígido es descrito por las ecuaciones siguientes:

$$H(q)\ddot{q} + V(q, \dot{q}) + G(q) = \tau$$

Donde  $H(q)$  es una matriz de inercias de  $n \times n$ ,  $q$  y sus estados derivativos son vectores de coordenadas generalizadas, y  $V(q, \dot{q})$ ,  $G(q)$ , y  $\tau$  son vectores de  $n \times 1$  que contienen torques dependientes de velocidad, torques de gravedad y torques de entrada respectivamente. Sus ecuaciones son descritas y estudiadas más adelante. Este ejemplo se centra, sobrescribiendo las  $n$  ecuaciones diferenciales acopladas dentro de la forma de espacio de estados, por lo tanto el vector de estados  $x$  es:

$$x = \begin{bmatrix} q \\ \dot{q} \end{bmatrix}$$

El vector de entrada es  $u = \tau$ , y tomando el vector de salida como  $y = q$ . Unas de las propiedades especiales de los robots rígidos que se discutirán más adelante, es que la matriz  $H(q)$  es conocida e invertible, quedando:

$$\begin{bmatrix} \dot{q} \\ \ddot{q} \end{bmatrix} = \begin{bmatrix} \dot{q} \\ -M^{-1}(V+G) \end{bmatrix} + \begin{bmatrix} 0 \\ M^{-1} \end{bmatrix} \tau$$

$$y = \begin{bmatrix} I & 0 \end{bmatrix} \begin{bmatrix} q \\ \dot{q} \end{bmatrix}$$

o

$$\dot{x} = F(x) + G(x)u = f(x, u)$$

$$y = \begin{bmatrix} I & 0 \end{bmatrix} x$$

donde

$$F(x) = \begin{bmatrix} \dot{q} \\ -M^{-1}(V+G) \end{bmatrix}, \quad G(x) = \begin{bmatrix} 0 \\ M^{-1} \end{bmatrix}$$

## 3.4.3 Conceptos de estabilidad.

## 3.4.3.1 Puntos de equilibrio.

El primer de concepto de estabilidad que veremos, para poder comprender el comportamiento de los sistemas libres o equivalentes, son los sistemas forzados con una entrada dada. En otras palabras, se verá la estabilidad de un *punto de equilibrio*<sup>1</sup>, con respecto a los cambios en las condiciones iniciales de los sistemas.

Haciendo  $x_e$  como el estado de equilibrio de sistemas no lineales, continuos y posiblemente variables en el tiempo:

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

esto es,  $f(x_e, t) = 0$ , donde  $x, f$  son vectores de  $n \times 1$ .

**EXEMPLO 3.4.3-1**

Considerando las ecuaciones de un robot rígido del ejemplo 3.4.2-2 tomando el torque de entrada como:  $\tau = -K_v \dot{q} - K_p q + V + G$ , donde  $K_p$  y  $K_v$ , son matrices con elementos diagonales diferentes de cero. Entonces el sistema en lazo cerrado está dado por:

$$\begin{bmatrix} \dot{q} \\ \ddot{q} \end{bmatrix} = \begin{bmatrix} \dot{q} \\ -K_v \dot{q} - K_p q \end{bmatrix}$$

El punto de equilibrio de este sistema es:

$$x_e = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

<sup>1</sup>Para poder analizar con más detalle todo lo referente a este tema, se puede consultar el Khalil H. K. *Nonlinear Systems*. Ed. Prentice Hall 1996, cap. 1.

## 3.4.3.2 Estabilidad de Lyapunov.

La teoría de la estabilidad de Lyapunov está relacionada con el comportamiento de sistemas no lineales no forzados, descritos por ecuaciones diferenciales

$$\dot{x}(t) = f[x(t), t], \quad t \geq 0, \quad x \in R^n \quad (3.4.3-1)$$

donde sin pérdida de generalidad, el origen es un punto de equilibrio de (3.4.3-1). Esto podría parecer que no es necesario pero más adelante apoyándose en algunas propiedades, esto será muy útil.

Hay dos razones muy importantes por las que la teoría de Lyapunov es necesaria. La primera es que esta teoría permite determinar la estabilidad de un punto de equilibrio particular sin resolver las ecuaciones diferenciales. Esto es muy importante para el estudio de ecuaciones diferenciales no lineales, es un gran ahorro de operaciones. La segunda razón para usar la teoría de Lyapunov, es que esta provee los resultados cualitativos para diferentes cuestiones de estabilidad, por ejemplo para la estabilización en el diseño de controladores de sistemas dinámicos no lineales.

Por lo tanto a continuación describimos la estabilidad de Lyapunov [3]:

$$\begin{array}{l} x = f(x) \quad x(t) \in R^n \quad f(0) = 0 \\ f: D \rightarrow R^n \quad D \in R^n \end{array}$$

$$\begin{array}{l} \text{Lipschiz.} \quad \|f(x_1) - f(x_2)\| \leq L \|x_1 - x_2\| \quad L = cte. \\ \forall x_1, x_2 \in D \end{array}$$

<sup>1</sup>Para poder analizar con más detalle todo lo referente a este tema, se puede consultar el Khalil H. K. *Nonlinear Systems*. Ed. Prentice Hall 1996, cap. 2. pág. 57.

Se dice que el punto de equilibrio  $x = 0$  es:

- i) ESTABLE: si  $\forall \xi > 0$  ;  $\exists \delta = \delta(\xi)$  tal que  $\|x(0)\| < \delta \Rightarrow \|x(t)\| < \xi$
- ii) INESTABLE: si no es estable
- iii) ASINTOTICAMENTE ESTABLE: si

a) es estable

b)  $\|x(t)\| \rightarrow 0, \quad t \rightarrow \infty$

En base a lo anterior se puede describir ahora el siguiente teorema de Lyapunov:

Sea  $V : D \rightarrow R$  donde  $D \in R^n$ , una función continuamente diferenciable, se dice que  $V$  es positiva definida, si:

- i)  $V(x) > 0 \quad \forall x \in D - \{0\}$
- ii)  $V(0) = 0$

Por lo tanto:

$$\begin{aligned} \dot{x} &= f(x) \\ f(0) &= 0 \end{aligned}$$

- i)  $\dot{V}(x) \leq 0 \Rightarrow x=0$  es estable
- ii) Si  $\dot{V}(x) < 0 \quad \forall x \in D - \{0\}, \Rightarrow x=0$  asintóticamente estable.

$$\text{donde } \dot{V}(x) = \frac{\partial V(x)}{\partial x} \cdot f(x) \leq 0$$

$x = 0$  es estable si  $\exists$  una función Lyapunov.

$x = 0$  es asintóticamente estable, si  $\exists$  una función Lyapunov

### 3.4.3.3 Teorema LaSalle.

Este teorema de LaSalle[3], analiza los puntos seminegativos definidos, en contraste con Lyapunov que analiza los puntos negativos definidos. Por lo tanto se tiene:

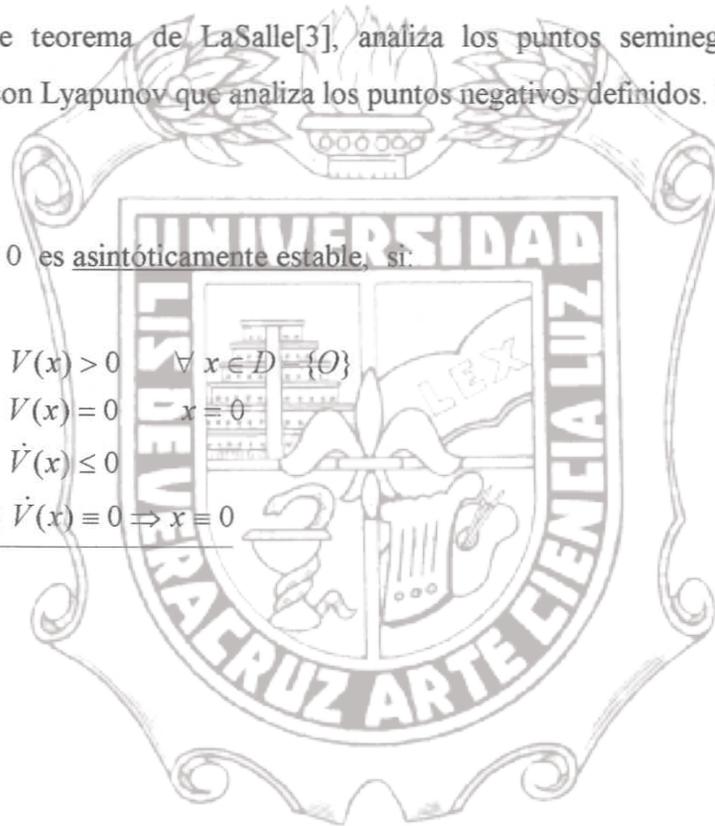
$x = 0$  es asintóticamente estable, si:

i)  $V(x) > 0 \quad \forall x \in D - \{0\}$

$V(x) = 0 \quad x = 0$

ii)  $\dot{V}(x) \leq 0$

iii)  $\dot{V}(x) = 0 \Rightarrow x = 0$



## IV. ANÁLISIS Y DISEÑO

## 4.1 Análisis.

Para poder empezar a analizar cualquier sistema, es necesario obtener su modelo matemático, una de las diferentes técnicas que se pueden aplicar es por el método de Lagrange [7], que obtienen las ecuaciones de movimiento para un sistema conservativo:

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = \tau$$

donde:

$q$  = n-vector de coordenadas generalizadas  $q_i$

$\tau$  = n-vector....  $\tau_i$

El lagrangiano es la diferencia entre la energía cinética y la potencial [8]:

$$L = K - P$$

En este caso:

$q$  = punto variable (vector), consistiendo de puntos ángulos  $\theta_i$  (grados o radianes) y puntos offsets<sup>1</sup>  $d_i$  (mts.).

Entonces  $\tau$  es un vector de componentes  $\tau_i$  de torque (newton-metros) correspondiente a los puntos de ángulos, y  $f_i$  de fuerza (newtons) correspondiente a los puntos de corrimiento (offsets).

<sup>1</sup> O puntos de corrimiento. Para analizar con más detalle consultar el Lewis F. L. *Control of Robots Manipulators*. Macmillan Publishing Company, 1993. Cap. 2, pág. 64.

## 4.1.1 Dinámica del brazo de 2 articulaciones (Estructura y propiedades) [7]:

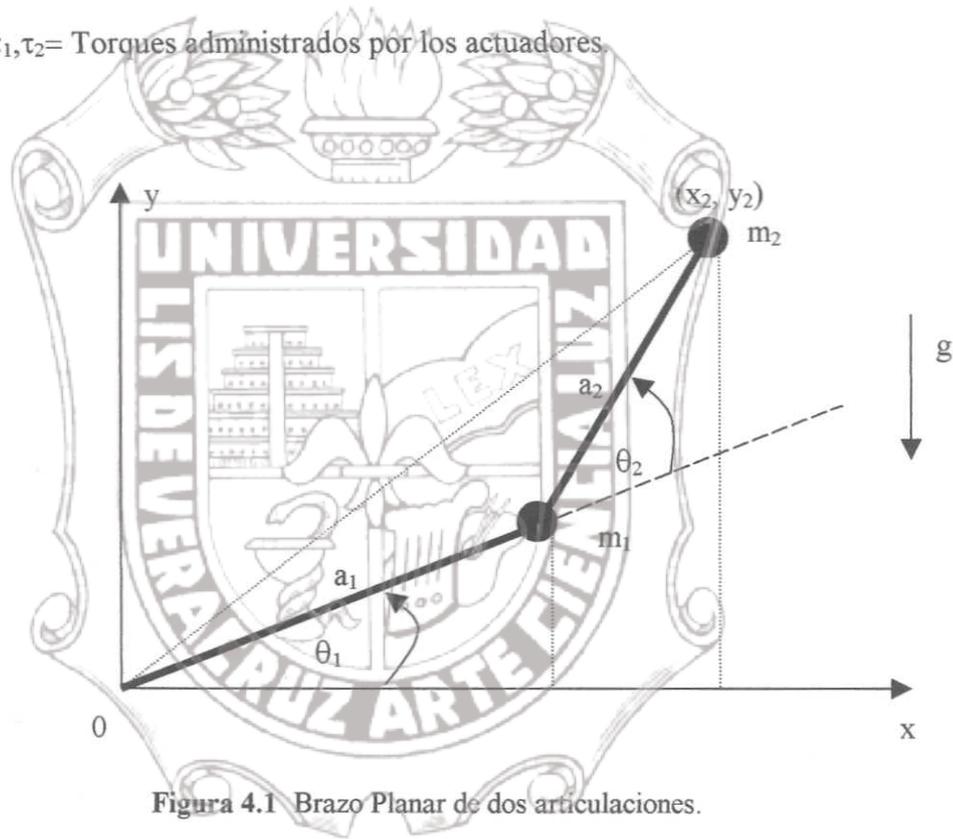
El punto variable:

$$q = [\theta_1 \theta_2]^T$$

El vector de fuerzas generalizadas es:

$$\tau = [\tau_1 \tau_2]^T$$

donde:  $\tau_1, \tau_2 =$  Torques administrados por los actuadores.



Desarrollando:

*Energía cinética y potencial:*

1ª Articulación:

$$K_1 = \frac{1}{2} m_1 a_1^2 \dot{\theta}_1^2$$

$$P_1 = m_1 g a_1 \text{sen} \theta_1$$

2ª Articulación:

$$x_2 = a_1 \cos \theta_1 + a_2 \cos (\theta_1 + \theta_2)$$

$$y_2 = a_1 \sin \theta_1 + a_2 \sin (\theta_1 + \theta_2)$$

$$\dot{x}_2 = -a_1 \dot{\theta}_1 \sin \theta_1 - a_2 (\dot{\theta}_1 + \dot{\theta}_2) \sin (\theta_1 + \theta_2)$$

$$\dot{y}_2 = a_1 \dot{\theta}_1 \cos \theta_1 + a_2 (\dot{\theta}_1 + \dot{\theta}_2) \cos (\theta_1 + \theta_2)$$

como:

∴

$$v_2^2 = a_1^2 \dot{\theta}_1^2 + a_2^2 (\dot{\theta}_1 + \dot{\theta}_2)^2 + 2a_1 a_2 (\dot{\theta}_1^2 + \dot{\theta}_1 \dot{\theta}_2) \cos \theta_2$$

La energía cinética para la 2ª articulación:

$$K_2 = \frac{1}{2} m_2 v_2^2 = \frac{1}{2} m_2 a_1^2 \dot{\theta}_1^2 + \frac{1}{2} m_2 a_2^2 (\dot{\theta}_1 + \dot{\theta}_2)^2 + m_2 a_1 a_2 (\dot{\theta}_1^2 + \dot{\theta}_1 \dot{\theta}_2) \cos \theta_2$$

La energía potencial:

$$P_2 = m_2 g y_2 = m_2 g [a_1 \sin \theta_1 + a_2 \sin (\theta_1 + \theta_2)]$$

∴

El Lagrangiano:

$$L = K - P = K_1 + K_2 - P_1 - P_2$$

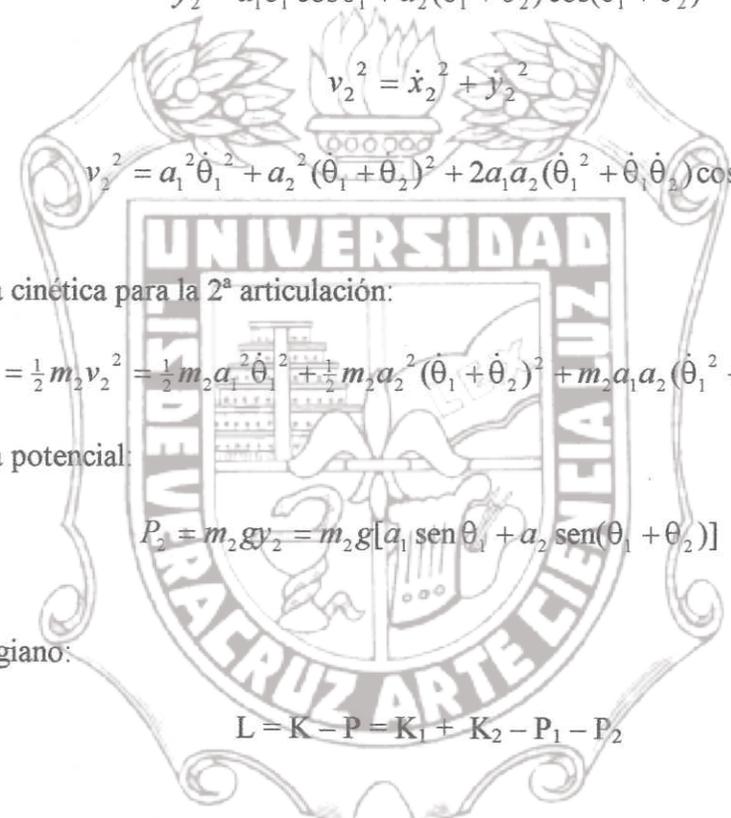
$$L = \frac{1}{2} (m_1 + m_2) a_1^2 \dot{\theta}_1^2 + \frac{1}{2} m_2 a_2^2 (\dot{\theta}_1 + \dot{\theta}_2)^2 + m_2 a_1 a_2 (\dot{\theta}_1^2 + \dot{\theta}_1 \dot{\theta}_2) \cos \theta_2 - (m_1 + m_2) g a_1 \sin \theta_1 - m_2 g a_2 \sin (\theta_1 + \theta_2)$$

Aplicando la ecuación de Lagrange:

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = \tau$$

Tenemos:

Instituto de Ingeniería  
 Universidad Veracruzana



$$\frac{\partial L}{\partial \dot{\theta}_1} = (m_1 + m_2)a_1^2 \dot{\theta}_1 + m_2 a_2^2 (\dot{\theta}_1 + \dot{\theta}_2) + m_2 a_1 a_2 (2\dot{\theta}_1 + \dot{\theta}_2) \cos \theta_2$$

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{\theta}_1} = (m_1 + m_2)a_1^2 \ddot{\theta}_1 + m_2 a_2^2 (\ddot{\theta}_1 + \ddot{\theta}_2) + m_2 a_1 a_2 (2\ddot{\theta}_1 + \ddot{\theta}_2) \cos \theta_2 - m_2 a_1 a_2 (2\dot{\theta}_1 \dot{\theta}_2 + \dot{\theta}_2^2) \operatorname{sen} \theta_2$$

$$\frac{\partial L}{\partial \theta_1} = -(m_1 + m_2) g a_1 \cos \theta_1 - m_2 g a_2 \cos (\theta_1 + \theta_2)$$

$$\frac{\partial L}{\partial \dot{\theta}_2} = m_2 a_2^2 (\dot{\theta}_1 + \dot{\theta}_2) + m_2 a_1 a_2 \dot{\theta}_1 \cos \theta_2$$

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{\theta}_2} = m_2 a_2^2 (\ddot{\theta}_1 + \ddot{\theta}_2) + m_2 a_1 a_2 \ddot{\theta}_1 \cos \theta_2 - m_2 a_1 a_2 \dot{\theta}_1 \dot{\theta}_2 \operatorname{sen} \theta_2$$

$$\frac{\partial L}{\partial \theta_2} = -m_2 a_1 a_2 (\dot{\theta}_1^2 + \dot{\theta}_1 \dot{\theta}_2) \operatorname{sen} \theta_2 - m_2 g a_2 \cos (\theta_1 + \theta_2)$$

Finalmente, de acuerdo con la ecuación de Lagrange, la dinámica del brazo está dada por las dos ecuaciones diferenciales no lineales [9] acopladas incluyendo fricción [propiedad 4.1.1-4].

$$\tau_1 = (m_1 + m_2)a_1^2 \ddot{\theta}_1 + m_2 a_2^2 (\ddot{\theta}_1 + \ddot{\theta}_2) + m_2 a_1 a_2 (2\ddot{\theta}_1 + \ddot{\theta}_2) \cos \theta_2 - m_2 a_1 a_2 (2\dot{\theta}_1 \dot{\theta}_2 + \dot{\theta}_2^2) \operatorname{sen} \theta_2 + (m_1 + m_2) g a_1 \cos \theta_1 + m_2 g a_2 \cos (\theta_1 + \theta_2)$$

∴

$$\tau_1 = \ddot{\theta}_1 [(m_1 + m_2)a_1^2 + m_2 a_2^2 + 2m_2 a_1 a_2 \cos \theta_2] + \ddot{\theta}_2 [m_2 a_2^2 + m_2 a_1 a_2 \cos \theta_2] - m_2 a_1 a_2 (2\dot{\theta}_1 \dot{\theta}_2 + \dot{\theta}_2^2) \operatorname{sen} \theta_2 + (m_1 + m_2) g a_1 \cos \theta_1 + m_2 g a_2 \cos (\theta_1 + \theta_2) + [v_1 \dot{\theta}_1 + k_1 \operatorname{sgn}(\dot{\theta}_1)]$$

$$\tau_2 = m_2 a_2^2 (\ddot{\theta}_1 + \ddot{\theta}_2) + m_2 a_1 a_2 \ddot{\theta}_1 \cos \theta_2 - m_2 a_1 a_2 \dot{\theta}_1 \dot{\theta}_2 \sin \theta_2 + m_2 a_1 a_2 (\dot{\theta}_1^2 + \dot{\theta}_1 \dot{\theta}_2) \sin \theta_2 + m_2 g a_2 \cos(\theta_1 + \theta_2)$$

∴

$$\tau_2 = \ddot{\theta}_1 [m_2 a_2^2 + m_2 a_1 a_2 \cos \theta_2] + m_2 a_2^2 \ddot{\theta}_2 + m_2 a_1 a_2 \dot{\theta}_1^2 \sin \theta_2 + m_2 g a_2 \cos(\theta_1 + \theta_2) + [v_2 \dot{\theta}_2 + k_2 \operatorname{sgn}(\dot{\theta}_2)]$$

Vectorialmente:

$$\begin{bmatrix} (m_1 + m_2) a_1^2 + m_2 a_2^2 + 2 m_2 a_1 a_2 \cos \theta_2 & m_2 a_2^2 + m_2 a_1 a_2 \cos \theta_2 \\ m_2 a_2^2 + m_2 a_1 a_2 \cos \theta_2 & m_2 a_2^2 \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} -m_2 a_1 a_2 (2 \dot{\theta}_1 \dot{\theta}_2 + \dot{\theta}_2^2) \sin \theta_2 \\ m_2 a_1 a_2 \dot{\theta}_1^2 \sin \theta_2 \end{bmatrix} + \begin{bmatrix} (m_1 + m_2) g a_1 \cos \theta_1 + m_2 g a_2 \cos(\theta_1 + \theta_2) \\ m_2 g a_2 \cos(\theta_1 + \theta_2) \end{bmatrix} + \begin{bmatrix} v_1 \dot{\theta}_1 + k_1 \operatorname{sgn}(\dot{\theta}_1) \\ v_2 \dot{\theta}_2 + k_2 \operatorname{sgn}(\dot{\theta}_2) \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \quad (4.1-1)$$

A continuación se investigan algunos detalles de la estructura y propiedades de las ecuaciones dinámicas de un brazo, en el caso de la estructura esta seriamente reflejada en la forma de una ley de control. El controlador es más simple y efectivo si conocemos las propiedades del brazo y son incorporadas en la plataforma de diseño.

En realidad, un brazo robótico es siempre afectado por la fricción y perturbaciones. Entonces podemos generalizar el modelo de un brazo derivado de las dinámicas del manipulador [7] como:

$$H(q)\ddot{q} + V(q, \dot{q}) + G(q) + D\dot{q} + \tau_d = \tau \quad (4.1-2)$$

donde:  $V(q, \dot{q}) = c(q, \dot{q})\dot{q}$

- $H(q)$  = Matriz de inercia (simétrica)
- $c(q, \dot{q})\dot{q}$  = Vector coriolis / centripeta =  $V(q, \dot{q})$
- $D\dot{q}$  = Vector de fricción =  $F(\dot{q})$
- $G(q)$  = Vector de gravedad
- $\tau_d$  = Perturbación

Para la representación en variables de estado se puede escribir de la siguiente manera:

$$\frac{d}{dt} \begin{bmatrix} q \\ \dot{q} \end{bmatrix} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix} \begin{bmatrix} q \\ \dot{q} \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} u$$

donde:  $u = H^{-1}(\tau - N)$   
 $N = V + G + F + \tau_d$

∴ si  $q = \theta$

$$\begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

donde:  $[u_1 \ u_2]^T = H^{-1}(\tau - N)$

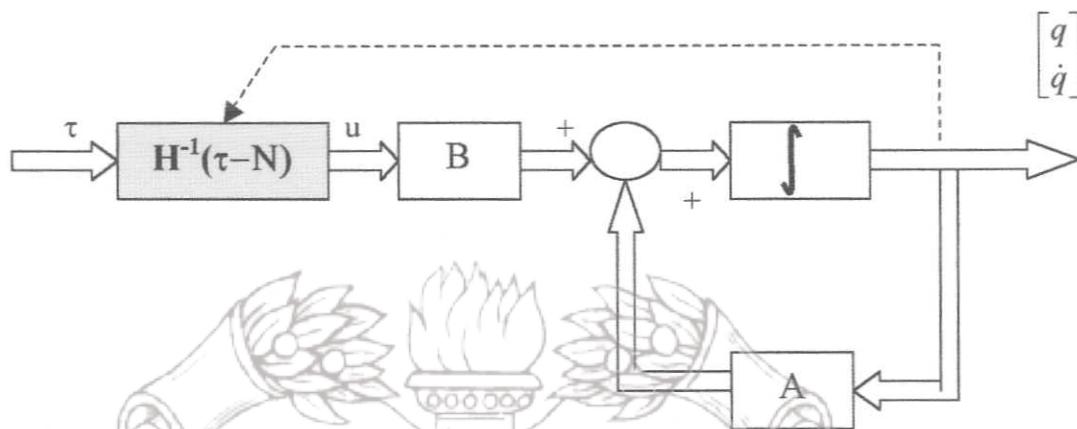


Figura 4.2 Representación en variables de estado para sistemas no lineales. El bloque sombreado contiene las no linealidades, que dependen del estado  $[q \ \dot{q}]^T$ , lo que se representa mediante la línea discontinua.

Recordando que:

$$\|q\| = (q^T q)^{1/2}$$

$$\|A\| \cong (\lambda_{\max}(A^T A))^{1/2}$$

**Propiedad No. 4.1.1-1.** [7] La matriz de inercia  $H(q)$  es simétrica positiva definida. La energía cinética del brazo es:

$$K = \frac{1}{2} \dot{q}^T H(q) \dot{q}$$

como  $K > 0$  a menos que  $\dot{q} \equiv 0$ . Se concluye que  $H(q) > 0$ . Algunas expresiones como  $\dot{H}$  están dadas mas adelante.

Otra propiedad vital de  $H(q)$  es que está limitada por encima y por debajo.

$$\mu_1 I \leq H(q) \leq \mu_2 I$$

Con  $\mu_1$  y  $\mu_2$  como escalares esto puede ser calculado para un brazo dado. Cuando se dice que  $\mu_1 I \leq H(q)$ , por instancia, se observa que esta matriz resultante es positiva semidefinida. Esto es:

donde:  $x \in R^n$

Por otro lado la matriz de inercia está limitada por:

$$\frac{1}{\mu_2} I \leq H^{-1}(q) \leq \frac{1}{\mu_1} I$$

Si el brazo fuera tipo  $V$  o de revolución, los límites  $\mu_1$  y  $\mu_2$  son constantes, dado que  $q$  sólo aparece en  $H(q)$  por medio de los términos  $\sin$  y  $\cos$  (seno y coseno respectivamente). Si el brazo es prismático, entonces  $\mu_1$  y  $\mu_2$  pueden ser funciones escalares de  $q$ . Por lo tanto los límites de la matriz de inercia pueden estar dados por:

$$m_1 \leq \|H(q)\| \leq m_2$$

donde algunas normas de matrices inducidas pueden ser usadas para definir los escalares positivos  $m_1$  y  $m_2$ .

**Propiedad No. 4.1.1-2.** [7] Propiedades de los términos Coriolis/Centripeta.

Para esto consideramos la ecuación de Lagrange:

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = \tau$$

donde:  $L(q, \dot{q}) = K(q, \dot{q}) - P(q) = \frac{1}{2} \dot{q}^T H(q) \dot{q} - P(q)$

Esta es una propiedad fundamental donde la energía cinética como ya se había mencionado es una función cuadrática de los puntos del vector de velocidad, y la energía potencial es independiente de  $\dot{q}$ .

Los términos requeridos en la ecuación de Lagrange, ahora son dados por:

$$\begin{aligned} \frac{\partial L}{\partial \dot{q}} &= \frac{\partial K}{\partial \dot{q}} = H(q) \dot{q} \\ \frac{d}{dt} \frac{\partial L}{\partial \dot{q}} &= H(q) \ddot{q} + \dot{H}(q) \dot{q} \\ \frac{\partial L}{\partial q} &= \frac{1}{2} \frac{\partial}{\partial q} (\dot{q}^T H(q) \dot{q}) - \frac{\partial P(q)}{\partial q} \end{aligned}$$

Entonces la ecuación dinámica del brazo es:

$$H(q) \ddot{q} + \dot{H}(q) \dot{q} - \frac{1}{2} \frac{\partial}{\partial q} (\dot{q}^T H(q) \dot{q}) + \frac{\partial P(q)}{\partial q} = \tau$$

Definiendo el vector *coriolis/centripeta* como:

$$V(q, \dot{q}) = \dot{H}(q) \dot{q} - \frac{1}{2} \frac{\partial}{\partial q} (\dot{q}^T H(q) \dot{q}) = \dot{H} \dot{q} - \frac{\partial K}{\partial q}$$

y el vector de gravedad:

$$G(q) = \frac{\partial P(q)}{\partial q}$$

ya que

$$P(q) = -\sum_{i=1}^n q^T T_i(q) I_i e_4$$

donde  $T_i$  es una transformación homogénea<sup>2</sup> de  $4 \times 4$ , y  $e_4$  es la última columna de una matriz identidad de  $4 \times 4$ . Todo esto revela un gran problema, si no se comprende, el estudio de las dinámicas del manipulador se pueden hacer muy confusas.

La simplificación del término  $V(q, \dot{q})$  requiere tomar los derivativos de una matriz [ejem.  $H(q)$ ] con respecto al  $n$ -vector  $q$ .

Para esto se considera el producto de Kronecker para el análisis de  $V(q, \dot{q})$ . Primero se examina el término de  $V$  desde el punto de vista del producto de Kronecker [7], definido por dos matrices  $A \in R^{n \times m}$ ,  $B \in R^{p \times q}$  como:

$$A \otimes B = [a_{ij} B] \in R^{np \times mq},$$

donde  $a_{ij}$  son los elementos de  $A$  y  $[a_{ij} B]$  implica un bloque matricial de  $np \times mq$  compuesto de los  $p \times q$  bloques  $a_{ij} B$ . Si para  $A \in R^{3 \times 3}$ , teniendo:

$$A \otimes B = \begin{bmatrix} a_{11}B & a_{12}B & a_{13}B \\ a_{21}B & a_{22}B & a_{23}B \\ a_{31}B & a_{32}B & a_{33}B \end{bmatrix}$$

<sup>2</sup>Para poder analizar con más detalle todo lo referente a este tema, se puede consultar el Lewis F. L. *Control of Robots Manipulators*. Macmillan Publishing Company, 1993, en el apéndice A.

Para las matrices  $A(q)$ ,  $B(q)$ , con  $q \in R^n$ , se define la matriz derivativa como:

$$\frac{\partial A}{\partial q} = \begin{bmatrix} \frac{\partial A}{\partial q_1} \\ \vdots \\ \frac{\partial A}{\partial q_n} \end{bmatrix}$$

Entonces se puede probar la regla del producto:

$$\frac{\partial}{\partial q} [A(q)B(q)] = (I_n \otimes A) \frac{\partial B}{\partial q} + \frac{\partial A}{\partial q} B,$$

con  $I_n$  identidad de  $n \times n$ .

Ahora se puede examinar el vector Coriolis/centrípeta  $V(q, \dot{q})$ . Usando la regla del producto en la ecuación calculada de  $V(q, \dot{q})$  se tiene:

$$V(q, \dot{q}) = \dot{H}(q)\dot{q} - \frac{1}{2} \left[ \frac{\partial}{\partial q} (\dot{q}^T H(q)) \right] \dot{q}$$

$$V(q, \dot{q}) = \dot{H}(q)\dot{q} - \frac{1}{2} (I_n \otimes \dot{q}^T) \frac{\partial H}{\partial q} \dot{q},$$

o

$$V(q, \dot{q}) = [\dot{H}(q) - \frac{1}{2} U(q, \dot{q})] \dot{q},$$

donde:

$$U(q, \dot{q}) \equiv (I_n \otimes \dot{q}^T) \frac{\partial M}{\partial q}$$

también se puede recordar que:

$$V(q, \dot{q}) = c(q, \dot{q})\dot{q}$$

con la matriz de coeficientes dada por:

$$c(q, \dot{q}) \equiv \dot{H} - \frac{1}{2}U.$$

Para encontrar la expresión equivalente de  $c$ , note esto:

$$\dot{H}(q) = \sum_{i=1}^n \frac{\partial H}{\partial q_i} \dot{q}_i$$

la cual se puede escribir como:

ó como:

(Note que  $\partial H/\partial q_i$  es simétrica). Entonces:

$$c(q, \dot{q}) = [(\dot{q}^T \otimes I_n) - \frac{1}{2}(I_n \otimes \dot{q}^T)] \frac{\partial H}{\partial q}$$

Considerando:

$$V(q, \dot{q}) = \left[ \frac{\partial H}{\partial q} \right]^T (\dot{q}^T \otimes I_n) - \frac{1}{2}(I_n \otimes \dot{q}^T) \frac{\partial H}{\partial q} \dot{q}$$

$$V(q, \dot{q}) = [U^T(q, \dot{q}) - \frac{1}{2}U(q, \dot{q})] \dot{q}$$

y

$$V(q, \dot{q}) = c(q, \dot{q})\dot{q}$$

entonces:

$$S(q, \dot{q}) \equiv \dot{H}(q) - 2c(q, \dot{q})$$

es *skew simétrica* (propiedad 4.1.1-5), con  $x^T S x = 0$  para toda  $x \in R^n$ . En base a lo antes visto, se puede definir:

$$c(q, \dot{q}) \equiv \frac{1}{2}(M + U^T - U)$$

pero:

$$S(q, \dot{q}) = U - U^T$$

Esta  $c(q, \dot{q})$  es la estándar usada en algoritmos modernos de control robustos y adaptivos.

**Propiedad No. 4.1.1-3.** [7] Propiedades de los términos de gravedad  $G(q)$ .

(a)

El vector  $G(q)$  satisface de acuerdo a la propiedad 4.1.1-2:

$$G(q) = \frac{\partial P}{\partial q} = - \sum_{i=1}^n \frac{\partial}{\partial q} (g^T T_i(q)) I_i e_4$$

utilizando:  $\frac{\partial}{\partial q} [A(q)B(q)] = (I_n \otimes A) \frac{\partial B}{\partial q} + \frac{\partial A}{\partial q} B$ , de la propiedad 4.1.1-2, se tiene:

$$G(q) = - \sum_{i=1}^n \frac{\partial}{\partial q} (g^T T_i(q)) I_i e_4$$

$$G(q) = -\sum_{i=1}^n (I_n \otimes g^T) \frac{\partial T_i}{\partial q} I_i e_4 .$$

Un límite sobre los términos de gravedad puede ser derivado por algún brazo robótico dado.



$$\|G(q)\| \leq \partial y, \quad \partial y > 0$$

Esto quiere decir que este vector está acotado para el caso rotacional. Para el caso prismático, depende de un límite.

(b)

El vector  $G(q)$  satisface:

$$\frac{\partial G(q)}{\partial q} \leq \alpha$$

Si todos los elementos del vector están acotados, el vector está acotado.

Demostración.

De la propiedad anterior se tiene que:

$$G_i = -\sum_{j=1}^n G \frac{\partial {}^o T_j}{\partial q_i} \quad {}^j \bar{r}_i = -\sum_{j=1}^n G U_{ji} \quad {}^j \bar{r}_j$$

Puesto que segundas derivadas parciales  $\partial/\partial q$  producen solo términos acotados se concluye la demostración.

△

(c)

El vector de términos gravitacionales  $G(q)$  satisface:

$$\|G(q_1) - G(q_2)\| \leq \alpha \|q_1 - q_2\| \quad \forall q_1, q_2 \in R^n$$

Demostración:

Del inciso (b) aplicando el teorema del valor medio:

El teorema dice: "...si  $G(q)$  es continua, diferenciable y sus derivadas parciales con respecto a  $q$  están acotadas, entonces para cualquier par de vectores  $q_1, q_2$  se tiene:

$$G(q_1) - G(q_2) = \frac{\partial G(q)}{\partial q} \Big|_{q_{12}} (q_1 - q_2),$$

donde  $q_{12}$  es un punto entre  $q_1$  y  $q_2$ ...", por lo tanto tomando normas:

$$\begin{aligned} \|G(q_1) - G(q_2)\| &= \left\| \frac{\partial G(q)}{\partial q} \Big|_{q_{12}} (q_1 - q_2) \right\| \\ &\leq \left\| \frac{\partial G(q)}{\partial q} \Big|_{q_{12}} \right\| \|q_1 - q_2\| \\ &\leq \alpha \|q_1 - q_2\| \end{aligned}$$

Δ

**Propiedad No. 4.1.1-4.** Propiedades de los términos de fricción [7]  $F(\dot{q})$ .

La fricción en la ecuación del robot manipulador (4.1-2) es de la forma:

$$F(\dot{q}) = F_v \dot{q} + F_d(\dot{q})$$

con  $F_v$  como matriz de coeficientes de fricción viscosa, y  $F_d$  un término de fricción dinámica. Los coeficientes de fricción son los parámetros más difíciles de determinar para un brazo dado, y  $F(\dot{q})$  representa solamente una aproximación matemática de su modelo<sup>3</sup>.

Cada fricción es un efecto local, nosotros podemos asumir que  $F(\dot{q})$  está desacoplado entre los puntos del brazo, como esto:

$$F(\dot{q}) = \text{vec}\{f_i(\dot{q}_i)\} = \begin{bmatrix} f_1(\dot{q}_1) \\ \vdots \\ f_n(\dot{q}_n) \end{bmatrix}$$

con  $f_i(\cdot)$  funciones escalares conocidas, estas pueden ser determinadas para un brazo dado. Hemos definido a la función  $\text{vec}\{\cdot\}$  para usarla más adelante.

La fricción viscosa puede ser asumida de la siguiente forma:

$$F_v \dot{q} = \text{vec}\{v_i \dot{q}_i\}$$

con  $v_i$  coeficientes constantes conocidos. Entonces  $F_v = \text{diag}\{v_i\}$ , como una matriz diagonal.

<sup>3</sup>Para adentrarse más en este tema, ver [Craig 1988, Schilling 1990].

La fricción dinámica puede ser asumida de la siguiente forma:

$$F_d(\dot{q}) = \text{vec}\{k_i \text{sign}(\dot{q}_i)\},$$

con  $k_i$  coeficientes constantes conocidos y la función signo definida para un escalar  $x$  por:

$$\text{sign}(x) = \begin{cases} +1, & x > 0 \\ \text{indeterminado}, & x = 0 \\ -1, & x < 0 \end{cases}$$

Entonces:

$$F_d(\dot{q}) = K_d \text{sign}(\dot{q}),$$

con  $K_d = \text{diag}\{k_i\}$  como la *matriz de coeficientes de fricción dinámica* y la función signo definida por un vector  $x$  por:

$$\text{sign}(x) = \text{vec}\{\text{sign}(x_i)\}.$$

Un límite sobre los términos de la fricción, puede ser asumido de la forma:

$$\|F_v \dot{q} + F_d(\dot{q})\| \leq v \|\dot{q}\| + k$$

con  $v$  y  $k$  conocidas para un brazo específico y  $\|\cdot\|$  como una norma.

**Propiedad No. 4.1.1-5.** La matriz *skew simétrica*  $N(q, \dot{q}) = \dot{H}(q) - 2c(q, \dot{q})$  es antisimétrica.

Demostración:

$$\text{Cada elemento de } \dot{H}(q) \text{ satisface } \dot{h}_{ij}(q) = \sum_{k=1}^n \frac{\partial h_{ij}}{\partial q_k} \dot{q}_k.$$

Por otro lado, de la obtención del modelo por medio de la traza<sup>4</sup>, utilizamos lo siguiente:

$$C_{ij} = \sum_{k=1}^n \frac{1}{2} \left( \frac{\partial_{ij}}{\partial q_k} + \frac{\partial_{ik}}{\partial q_j} - \frac{\partial_{kj}}{\partial q_i} \right) q_k$$

Por lo tanto:

$$h_{ij} = \sum_{k=1}^n \left( \frac{\partial h_{ij}}{\partial q_k} \frac{\partial h_{ij}}{\partial q_k} - \frac{\partial h_{ik}}{\partial q_j} \frac{\partial h_{kj}}{\partial q_i} \right) \dot{q}_k$$

$$= \sum_{k=1}^n \left( \frac{\partial h_{kj}}{\partial q_i} \frac{\partial h_{ik}}{\partial q_j} \right) \dot{q}_k$$

Esto significa que:

$$h_{ij} = \sum_{k=1}^n \left( \frac{\partial h_{ki}}{\partial q_j} \frac{\partial h_{jk}}{\partial q_i} \right) \dot{q}_k$$

$$= \sum_{k=1}^n \left( \frac{\partial h_{ik}}{\partial q_j} \frac{\partial h_{kj}}{\partial q_i} \right) \dot{q}_k$$

$$= \sum_{k=1}^n \left( \frac{\partial h_{kj}}{\partial q_i} \frac{\partial h_{ik}}{\partial q_j} \right) \dot{q}_k = -h_{ij}$$

Debe notarse que esta propiedad es válida gracias a la definición de  $c(q, \dot{q})$ , con base en los símbolos de Christoffel. Sin embargo, es posible encontrar otras definiciones de  $c(q, \dot{q})$  tales que  $V(q, \dot{q}) = c(q, \dot{q})\dot{q}$ . En este caso, esta propiedad ya no será válida. En cambio, siempre se cumple que:

$$\dot{q}^T (\dot{H}(q) - 2c(q, \dot{q}))\dot{q} = 0$$

<sup>4</sup>Para poder analizar con más detalle todo lo referente a este tema, se puede consultar el Fu K. S. *ROBOTICA-Control, detección, visión e inteligencia*. Ed. McGraw-Hill / Interamericana, 1988, cap. 3.

**Propiedad No. 4.1.1-6.** La matriz  $H'(q)$  existe y satisface:

$$\lambda \min(H(q))\|y\|^2 \leq y^T H(q)y \leq \lambda \max(H(q))\|y\|^2 \quad \forall y \in R^n$$

Demostración:

Puesto que  $H(q)$  es positiva definida, cada vector  $y \in R^n$  puede ser expresado en términos de una base ortonormal  $(y_1, \dots, y_n)$  como:

$$y = c_1 y_1 + \dots + c_n y_n$$

Satisfaciendo:

$$y^T H(q)y = c_1^2 \lambda_1(H(q)) + \dots + c_n^2 \lambda_n(H(q))$$

$$y^T y = \|y\|^2 = c_1^2 + \dots + c_n^2$$

Δ

**Propiedad No. 4.1.1-7.** La matriz  $H'(q)$  existe y satisface:

$$\lambda^{-1} \max(H(q))\|y\|^2 \leq y^T H^{-1}(q)y \leq \lambda^{-1} \min(H(q))\|y\|^2 \quad \forall y \in R^n$$

Demostración:

Se sigue directamente de la propiedad anterior.

**Propiedad No. 4.1.1-8.** La matriz  $H^{-1}(q)$  existe y satisface:

$$0 < \sigma_n \leq \|H^{-1}(q)\| \leq \sigma_H < \infty$$

Demostración:

Igual a la demostración anterior pero con:

$$\sigma_n = \min_{q \in R^n} \lambda^{-1} \max(H(q)) \quad , \quad \sigma_H = \max_{q \in R^n} \lambda^{-1} \min(H(q))$$

Esta propiedad es útil para Lyapunov. Δ

**Propiedad No. 4.1.1-9.** Propiedad relacionada con el modelo. Mediante una definición apropiada de sus parámetros, el modelo (4.1-2) puede escribirse como:

$$H(q)\ddot{q} + c(q, \dot{q})\dot{q} + D\dot{q} + G(q) = Y(q, \dot{q}, \ddot{q}) \phi = \tau$$

donde  $Y(q, \dot{q}, \ddot{q})$  es el regresor y  $\phi$  es el vector de parámetros.

## 4.2. Diseño

Para esto se toman algunas propiedades antes mencionadas, además de utilizar diferentes herramientas de apoyo para el diseño de controladores aplicados a sistemas no lineales, como retroalimentación de estados, Lyapunov, etc.

### 4.2.1. Par calculado.

Para el diseño de todos los controladores, se toma la forma general del modelo dinámico de un robot planar de  $n$  grados de libertad (4.1-2), pero se maneja de la siguiente forma:

$$H(q)\ddot{q} + V(q, \dot{q}) + g(q) = \tau \quad (4.2-1)$$

donde:

$$V(q, \dot{q}) = c(q, \dot{q})\dot{q} + D\dot{q}$$

$$g(q) = G(q)$$

Para cuestiones de facilidad, se puede asumir que  $\tau$  se puede elegir libremente (no se toma la dinámica de los actuadores). Diseñando por retroalimentación de estados se puede establecer que para el seguimiento de la trayectoria se puede elegir:

$$\tau = H(q)\ddot{q}_d + c(q, \dot{q})\dot{q} + D\dot{q} + g(q)$$

donde:  $q_d, \dot{q}_d, \ddot{q}_d$  son señales deseadas.

En lazo cerrado tendríamos:

$$H(q)\ddot{q} = H(q)\ddot{q}_d$$

y como  $H(q)$  es invertible de acuerdo a las propiedades (4.1.1-6), (4.1.1-7) y (4.1.1-8), se tiene:

$$\ddot{q} = \ddot{q}_d$$

El sistema anterior producirá un buen seguimiento solo si las condiciones iniciales de las señales deseadas y de las variables de las articulaciones son iguales.

Por lo que para corregir este problema, diseñamos el siguiente controlador.

Ya que nuestro objetivo de control es:

$$e = q_d - q \rightarrow 0 \quad \text{cuando } t \rightarrow 0 \quad (4.2-2)$$

∴

$$q = q_d - e$$

$$\ddot{q} = \ddot{q}_d - \ddot{e}$$

y nuestra dinámica del error es por lo tanto:

$$\ddot{e} + K_v \dot{e} + K_p e = 0 \quad (4.2-3)$$

Entonces se tiene:

$$\tau = H(q)(\ddot{q}_d + K_v \dot{e} + K_p e) + c(q, \dot{q})\dot{q} + D\dot{q} + g(q) \quad (4.2-4)$$

o

$$\tau = \underbrace{H(q)(K_v \dot{e} + K_p e)}_{t_{fb}} + \underbrace{H(q)\ddot{q}_d + c(q, \dot{q})\dot{q} + D\dot{q} + g(q)}_{t_{ff}}$$

donde:

$t_{fb}$  = Es la componente de retroalimentación, que provee la cantidad de momento necesaria para deducir los errores de seguimiento de trayectoria.

$t_{ff}$  = Se le conoce como el término predictivo y sirve para proveer la cantidad de momento necesario para llevar al sistema a lo largo de su trayectoria nominal.

### Demostración.

La ecuación (4.2-3) la podemos escribir así:

$$\frac{d}{dt} \begin{bmatrix} e \\ \dot{e} \end{bmatrix} = \begin{bmatrix} 0 & I \\ -K_p & -K_v \end{bmatrix} \begin{bmatrix} e \\ \dot{e} \end{bmatrix}$$

Para verificar que esta ecuación es válida se tiene que demostrar que los valores propios de A tienen parte real negativa.

Sea  $\lambda$  un valor propio de A, y si  $V = [v_1 \ v_2]^T \neq 0$  el vector propio asociado, entonces por definición:

$$\lambda \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} 0 & I \\ -K_p & -K_v \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} v_2 \\ -K_p v_1 - K_v v_2 \end{bmatrix}$$

Por lo tanto cuando  $\lambda=0$ , no puede ser un valor propio porque automáticamente:

$$v_2 = 0 \text{ y por consiguiente } -K_p v_1 = 0 \Rightarrow v_1 = 0.$$

Para averiguar el signo de la parte real de  $\lambda$ , se calcula suponiendo que  $\|v1\|=1$ , lo siguiente:

$$\lambda^2 = \lambda^2 v1^* v1 = v1^* \lambda^2 v1$$

donde: \* = Transpuesto conjugado en caso de complejo.

$$\begin{aligned} \lambda^2 &= v1^* \lambda v2 \\ \lambda^2 &= v1^* (-Kp v1 - Kv v2) \\ \lambda^2 &= -v1^* Kp v1 - \lambda v1^* Kv v1 \\ \lambda^2 &= -\beta - \lambda \alpha \end{aligned}$$

con  $\beta, \alpha > 0$ . Entonces se tiene:

$$\lambda^2 + \lambda \alpha + \beta = 0$$

Por lo tanto la parte real de  $\lambda$  es negativa.

Δ

#### 4.2.2 Proporcional-Derivativo (PD).

Dado el sistema:

$$H(q)\ddot{q} + c(q, \dot{q})\dot{q} = \tau$$

y

$$e = q_d - q$$

⇒

$$q = q_d - e$$

∴

$$H(q)\ddot{e} + c(q, \dot{q})\dot{e} = \tau$$

$$H(q)\ddot{e} = -c(q, \dot{q})\dot{e} + \tau$$

(4.2-5)

se propone la siguiente función de Lyapunov:

$$V(q, \dot{q}) = \frac{1}{2} \dot{e}^T H(q) \dot{e} + \frac{1}{2} e^T Kp e$$

(4.2-6)

Derivando:

$$\dot{V}(q, \dot{q}) = \frac{1}{2} \dot{e}^T \dot{H}(q) \dot{e} + \dot{e}^T Kp e + \dot{e}^T H(q) \ddot{e}$$

(4.2-7)

sustituyendo (4.2-5) en (4.2-7), se tiene:

$$\begin{aligned} &= \frac{1}{2} \dot{e}^T \dot{H}(q) \dot{e} - \dot{e}^T c(q, \dot{q}) \dot{e} + \dot{e}^T \tau + \dot{e}^T K p e \\ &= \frac{1}{2} \dot{e}^T (\dot{H}(q) \dot{e} - 2c(q, \dot{q})) \dot{e} + \dot{e}^T \tau + \dot{e}^T K p e \end{aligned}$$

Aplicando la propiedad (4.1.1-5) quedaría:

$$= \dot{e}^T \tau + \dot{e}^T K p e$$

Proponiendo  $\tau$  de tal forma que  $\dot{V} < 0$ :

$$\begin{aligned} \dot{V} &= \dot{e}^T (-K v \dot{e} - K p e) + \dot{e}^T K p e \\ \dot{V} &= -\dot{e}^T K v \dot{e} \end{aligned}$$

Como la función anterior es solo seminegativa definida, se tiene que utilizar el teorema de LaSalle (3.6.3.3) para demostrar la estabilidad asintótica.

El conjunto S para el cual  $\dot{V} = 0$  está dado por:

$$S = \{ (q, \dot{q}); \dot{q} \equiv 0 \}$$

y para  $\dot{q}_d \equiv 0$ , el sistema en lazo cerrado está dado por:

$$H(q) \ddot{q} + c(q, \dot{q}) \dot{q} + k v \dot{q} + K p (q_d - q) = 0$$

asumiendo que  $q_d \equiv 0$  en lo anterior, se tiene:

$$K v \dot{q} = 0$$

Puesto que  $K p > 0$ , se sigue que  $q = 0$  es el máximo conjunto invariante en S tal que  $\dot{V} = 0$ . Por lo tanto el sistema es asintóticamente estable. Entonces  $\tau$  está dado por:

$$\tau = -K v \dot{e} - K p e \tag{4.2-8}$$

Para llevar a cabo el seguimiento de la trayectoria, considérese nuevamente el sistema (4.2-1) pero ahora sustituyendo:

$$\ddot{q} = \ddot{q}_d - \ddot{e}$$

$$\begin{aligned} H(q)(\ddot{q}_d - \ddot{e}) + c(q, \dot{q})(\dot{q}_d - \dot{e}) + D(\dot{q}_d - \dot{e}) + g(q) &= \tau \\ H(q)\ddot{e} + c(q, \dot{q})\dot{e} + D\dot{e} &= -\tau + [H(q)\ddot{q}_d + c(q, \dot{q})\dot{q}_d + D\dot{q}_d + g(q)] \\ H(q)\ddot{e} &= -c(q, \dot{q})\dot{e} - D\dot{e} + [-\tau + H(q)\ddot{q}_d + c(q, \dot{q})\dot{q}_d + D\dot{q}_d + g(q)] \end{aligned} \quad (4.2-9)$$

Proponiendo una función de Lyapunov:

$$V(\dot{q}, q) = \frac{1}{2} \dot{e}^T H(q) \dot{e} + \frac{1}{2} e^T K_p e$$

derivando:

$$\dot{V} = \frac{1}{2} \dot{e}^T \dot{H}(q) \dot{e} + \dot{e}^T K_p e + \dot{e}^T H(q) \ddot{e}$$

sustituyendo (4.2-9):

$$\dot{V} = \frac{1}{2} \dot{e}^T \dot{H}(q) \dot{e} + \dot{e}^T [-c(q, \dot{q})\dot{e} - D\dot{e} + [-\tau + [H(q)\ddot{q}_d + c(q, \dot{q})\dot{q}_d + D\dot{q}_d + g(q)]] + \dot{e}^T K_p e$$

$$\dot{V} = \frac{1}{2} \dot{e}^T [\dot{H}(q) - 2c(q, \dot{q})] \dot{e} + \dot{e}^T [-D\dot{e} - K_v \dot{e} - K_p e] + \dot{e}^T K_p e$$

$$\dot{V} = -\dot{e}^T K_{DV} \dot{e}$$

Por lo tanto por LaSalle tenemos que:  $K_{DV} > 0 \Rightarrow \dot{e} = 0 \Rightarrow \dot{V} = 0$

$\therefore \tau$  quedaría:

$$-\tau + H(q)\ddot{q}_d + c(q, \dot{q})\dot{q}_d + D\dot{q}_d + g(q) = -K_{DV}\dot{e} - K_p e$$

$\therefore$

$$\tau = H(q)\ddot{q}_d + c(q, \dot{q})\dot{q}_d + D\dot{q}_d + g(q) + K_{DV}\dot{e} + K_p e$$

## 4.2.3 Proporcional-Integral-Derivativo (PID por calculado).

Se tiene que ver que el control PD es muy efectivo si todos los parámetros del brazo son conocidos y no hay perturbaciones. También, desde la teoría del control clásico esto en presencia de perturbaciones, el control PD da un error en estado estable diferente de cero. Consecuentemente, esto nos motiva a diseñar un sistema tipo I incluyendo un integrador en el lazo de retroalimentación generando el control PID[10].

donde:

$$\dot{\epsilon} = \dot{e}$$

$$u = k_v \dot{e} + k_p e + k_i \epsilon$$

quedando:

$$\tau = \underbrace{H(q)(K_v \dot{e} + K_p e + K_i \epsilon)}_{t_{fb}} + \underbrace{H(q)\ddot{q}_a}_{t_{ff}} + c(q, \dot{q})\dot{q} + D\dot{q} + g(q)$$

donde:

$t_{fb}$  = Es la componente de retroalimentación, que provee la cantidad de momento necesaria para deducir los errores de seguimiento de trayectoria.

$t_{ff}$  = Se le conoce como el término predictivo y sirve para proveer la cantidad de momento necesario para llevar al sistema a lo largo de su trayectoria nominal.

Δ

## 4.2.4 Control basado en la Estructura Pasiva de los robots rígidos.

### 4.2.4.1 Introducción.

Este tipo de control como su nombre lo dice, está relacionado directamente con la estructura pasiva de los robots rígidos descritos en la ecuación (4.1-2), donde la matriz *skew simétrica* (propiedad 4.1.1-5) se considera aquí al igual que en el PD y que en los siguientes controladores.

Basado en la propiedad de la pasividad y el teorema de pequeña ganancia<sup>5</sup>, si se puede cerrar el lazo desde  $\dot{q}$  hasta  $\tau$  para un sistema pasivo (a lo largo de  $L_2$  con la entrada limitada) como se muestra en la figura 4.3, donde el sistema en lazo cerrado será *asintóticamente estable* [3] usando el teorema de pasividad. Nótese que esta entrada  $u_2$  está dada por un GDL extra para satisfacer el criterio de desarrollo. En otras palabras, eligiendo diferente  $L_2$  limitando  $u_2$  podemos ser capaces de obtener un mejor seguimiento de las trayectorias u obtener una inmunidad al ruido, lo que sería muy bueno.

Esta estructura mostrará la estabilidad asintótica de  $\dot{e}$  pero para  $e$  solo la estabilidad de Lyapunov. Por otro lado, si uno puede mostrar la pasividad del sistema, el cual mapea  $\tau$  hasta un nuevo vector  $s$ , el cual a su vez es una versión filtrada de  $e$ , un controlador cierra el lazo entre  $-s$  y  $\tau$  lo cual garantizará la estabilidad asintótica de ambos,  $e$  y  $\dot{e}$ .

<sup>5</sup>Este uso indirecto de la propiedad de pasividad y el teorema de la pequeña ganancia, está descrito en *Apuntes de control III*, cap. 4 Dr. Yu Tang Xu, UNAM. Marzo del 2000; pero para más detalles en Khalil H. K. *Nonlinear Systems*. Cap. 8. Ed. Prentice Hall 1996.

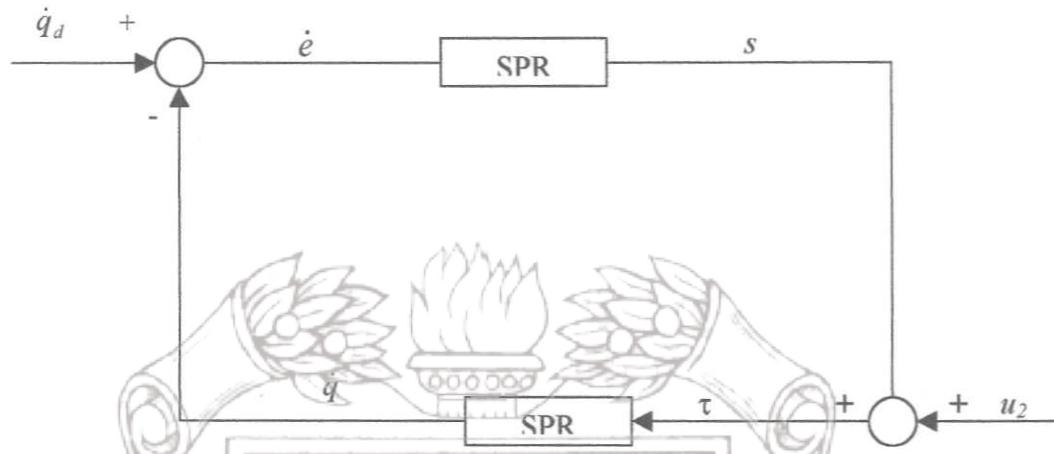


Figura 4.3 Control de estructura pasiva.

4.2.4.2 Diseño.

Considérese:

$$\begin{aligned}
 e &= q_d - q \\
 \dot{q}_r &= \dot{q}_d + \Lambda e \\
 s &= \dot{q}_r - \dot{q} = \dot{q}_d + \Lambda e - \dot{q} = \dot{e} + \Lambda e \\
 \text{con } \Lambda &> 0 \\
 \therefore \text{ si } s &\equiv 0 \Rightarrow \dot{e} + \Lambda e = 0 \\
 \therefore \dot{e} &= -\Lambda e
 \end{aligned}$$

En este caso  $s$  es un término que actúa sobre la matriz de regresión y que evita la posibilidad de utilizar derivadas en esta.

Ahora, la forma (4.1-2) la ponemos en términos de  $S$  tenemos:

$$\begin{aligned}
 H(q)(\ddot{q}_r - \dot{s}) + c(q, \dot{q})(\dot{q}_r - s) + F(\dot{q}) + g(q) &= \tau \\
 \therefore \\
 -[H(q)\dot{s} + c(q, \dot{q})s] + H(q)\ddot{q}_r + c(q, \dot{q})\dot{q}_r + F(\dot{q}) + g(q) &= \tau \\
 \therefore \\
 H(q)\dot{s} + c(q, \dot{q})s &= -\tau + H(q)\ddot{q}_r + c(q, \dot{q})\dot{q}_r + F(\dot{q}) + g(q)
 \end{aligned}$$

$$H(q)s = -\tau + Nr - c(q, \dot{q})s \tag{4.2-10}$$

donde:

$$Nr = H(q)\dot{q}_r + c(q, \dot{q})\dot{q}_r + F(\dot{q}) + g(q)$$

Se propone la siguiente función de Lyapunov:

$$V = \frac{1}{2} s^T H(q) s$$

derivando:

$$\dot{V} = \frac{1}{2} s^T \dot{H}(q) s + s^T H(q) \dot{s}$$

Sustituyendo (4.2-10):

$$\begin{aligned}
 &= \frac{1}{2} s^T \dot{H}(q) s + s^T [-\tau + Nr - c(q, \dot{q})s] \\
 &= \frac{1}{2} s^T [\dot{H}(q) - 2c(q, \dot{q})] s + s^T [-\tau + Nr] \\
 \therefore \\
 \dot{V} &= -s^T K_v s
 \end{aligned}$$

Pero como esta función es negativa semidefinida necesitamos determinar la estabilidad y por LaSalle:

$$\lim_{t \rightarrow \infty} s = 0$$

Lo que implica que  $e$  y  $\dot{e} \rightarrow 0$ , por lo tanto:

$$-\tau + Nr = -K_v s$$

$$\tau = Nr + K_v s$$

donde:  $Nr = H(q)\ddot{q}_r + c(q, \dot{q})\dot{q}_r + F(\dot{q}) + g(q)$

#### 4.2.5 Robusto Adaptable.

##### 4.2.5.1 Introducción.

El problema de diseñar leyes de control adaptable para robots manipuladores rígidos está en asegurar el seguimiento asintótico de la trayectoria, esto ha sido de interés por muchos años. El desarrollo de efectivos controladores adaptivos representa un importante paso para aplicaciones robóticas de alta-velocidad/presición. Para un buen desempeño de algunas aplicaciones de estos manipuladores en la industria, estos pueden depender de ciertos parámetros de las propiedades dinámicas de estos robots (ej. Momentos de inercia desconocidos, masas, distancias, fricciones, etc.). También estos parámetros son difíciles de calcular o medir, por ejemplo estos limitan el potencial para manejar objetos de considerable tamaño y peso.

Algunos controladores como el PD, utilizan la propiedad de la conservación de la energía pero esto tiene dos desventajas. Primero, el controlador simplemente asegura que el manipulador alcance el “set-point” (punto deseado). En general, un diseñador de controles para robots asegura que el manipulador siga una trayectoria deseada en tiempo variante (“time-varying”). Segundo, el controlador requiere exactamente el conocimiento de algunos parámetros asociados con el modelo del robot manipulador, donde los términos de gravedad y fricción son incluidos en las leyes de control.

4.2.5.2 Diseño:

Para resolver el problema de no conocer los parámetros, se diseña el control adaptable, seleccionando una función de Lyapunov, que está en función del error de seguimiento y el error paramétrico, pero antes se definen ciertas propiedades:

$$\begin{aligned}
 e &= q_d - q \\
 \dot{q}_r &= \dot{q}_d + \Lambda e \\
 s &= \dot{q}_r - \dot{q} = \dot{q}_d + \Lambda e - \dot{q} = \dot{e} + \Lambda e \\
 &\text{con } \Lambda > 0 \\
 \therefore \text{ si } s \equiv 0 &\Rightarrow \dot{e} + \Lambda e = 0 \\
 \therefore \dot{e} &= -\Lambda e
 \end{aligned}$$

$$\Gamma = \text{diag}(\gamma_1, \gamma_2, \dots, \gamma_r) \quad \text{donde: } \Gamma \in \mathbb{R}^{r \times r}; \gamma_r \in \mathbb{R}$$

$$\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n) \quad \text{donde: } \Lambda \in \mathbb{R}^{n \times n}; \lambda_n \in \mathbb{R}$$

$$\tilde{\varphi} = \varphi - \hat{\varphi} \quad \text{error paramétrico, donde: } \varphi \in \mathbb{R}^{m \times 1}$$

A  $s = \dot{e} + \Lambda e$  se puede considerar como un error de seguimiento filtrado.

Ahora si, en base a lo anterior se propone la función de Lyapunov.

$$V = \frac{1}{2} s^T H(q) s + \frac{1}{2} \tilde{\varphi}^T \Gamma \tilde{\varphi}$$

derivando:

$$\dot{V} = s^T H(q) \dot{s} + \frac{1}{2} s^T \dot{H}(q) s + \tilde{\varphi}^T \Gamma \dot{\tilde{\varphi}}$$

Pero antes se necesita  $\dot{s}$  para sustituirlo en  $\dot{V}$ , por lo que antes se define:

$$H(q)\ddot{q} + c(q, \dot{q})\dot{q} + F(\dot{q}) + g(q) = \tau = Y(q, \dot{q}, \ddot{q})\varphi$$

Y poniendo lo anterior en términos de  $s$  se tiene:

$$H(q)(\ddot{q}_r - \dot{s}) + c(q, \dot{q})(\dot{q}_r - s) + F(\dot{q}) + g(q) = \tau = Y(q, \dot{q}, \ddot{q}_r, \ddot{q}_r) \phi$$

Ahora se sustituye:

$$\dot{q}_r = \dot{q}_d + \Lambda e$$

$$-[H(q)\dot{s} + c(q, \dot{q})s] + H(q)(\ddot{q}_d + \Lambda \dot{e}) + c(q, \dot{q})(\dot{q}_d + \Lambda e) + F(\dot{q}) + g(q) = \tau = Ya\phi$$

$\therefore$

$$H(q)\dot{s} + c(q, \dot{q})s - F(\dot{q}) = H(q)(\ddot{q}_d + \Lambda \dot{e}) + c(q, \dot{q})(\dot{q}_d + \Lambda e) + g(q) = Ya\phi = \tau$$

$$H(q)\dot{s} = -\tau - c(q, \dot{q})s + Ya\phi + F(\dot{q})$$

(4.2-11)

$\therefore$

$Ya$  es una matriz  $n \times r$  de funciones conocidas en el tiempo. Esta es algún tipo de separación paramétrica, además notese que no está en función de  $\ddot{q}$ .

Volviendo a la función  $\dot{V}$ , sustituimos  $H(q)\dot{s}$ :

$$\dot{V} = \frac{1}{2} s^T \dot{H}(q)s + \tilde{\phi}^T \Gamma \tilde{\phi} + s^T [Ya\phi - \tau - c(q, \dot{q})s + F(\dot{q})]$$

$$= \frac{1}{2} s^T [\dot{H}(q) - 2c(q, \dot{q})]s + \tilde{\phi}^T \Gamma \tilde{\phi} + s^T [Ya\phi - \tau + F(\dot{q})]$$

de acuerdo a la propiedad (4.1.1-5), se tiene:

$$= \tilde{\phi}^T \Gamma \tilde{\phi} + s^T [Ya\phi - \tau + F(\dot{q})] = -s^T K_v s + \tilde{\phi}^T \Gamma \tilde{\phi}$$

Eligiendo  $\tau$  para que  $\dot{V} \leq 0$  se tiene:

$$\tau = Ya \hat{\phi} + F(\dot{q}) + K_v s$$

pero:

$$\dot{V} = s^T [Ya\phi - Ya\hat{\phi} - K_V s + F(\dot{q}) - F(\dot{q})] + \tilde{\phi}^T \Gamma \dot{\tilde{\phi}}$$

como  $\tilde{\phi} = \phi - \hat{\phi}$

$$= s^T [Ya\tilde{\phi} - K_V s] + \tilde{\phi}^T \Gamma \dot{\tilde{\phi}}$$

$$= -s^T K_V s + \underbrace{\tilde{\phi}^T [\Gamma \dot{\tilde{\phi}} + Ya s]}_0$$

Por lo tanto como  $\dot{V} \leq 0$  a todo el término positivo anterior se iguala a cero quedando:

$$\dot{V} = -s^T K_V s \leq 0$$

Ahora se verifica el tipo de estabilidad del error de seguimiento. Primero, como  $\dot{V} \leq 0$  se analiza por LaSalle, y se ve que:

$$K_V > 0 \text{ concluyendo que } s \rightarrow 0 \quad \therefore e \rightarrow 0 \text{ y } \dot{e} \rightarrow 0.$$

Además, puesto que  $V > 0$  y  $\dot{V} \leq 0$  se concluye que:

$$\|\tilde{\phi}\| < \infty \Rightarrow \|\hat{\phi}\| < \infty$$

Por lo tanto:

$$\Gamma \dot{\tilde{\phi}} = -Ya s$$

$$\dot{\tilde{\phi}} = -\Gamma^{-1} Ya s$$

$$\hat{\phi} = -\tilde{\phi} = \Gamma^{-1} Ya s$$

4.2.6 Robusto. Con compensación a la incertidumbre paramétrica, o control tipo "saturación".

4.2.6.1 Diseño.

De acuerdo al control anterior:

$$\begin{aligned} s &= \dot{e} + \Lambda e = \dot{q}_r - \dot{q} \\ \dot{q}_r &= \dot{q}_d + \Lambda(q_d - q) \end{aligned}$$

El adaptable se puede escribir:

$$\tau = Y\alpha \hat{\phi} + F(\dot{q}) + K_v s$$

Rescribiendo la compensación:

$$\tau = F(\dot{q}) + K_v s + Y\alpha(\hat{\phi} + w) \quad (5.2-12)$$

tomando la ecuación (5.2-11) del control anterior y sustituyendo  $\tau$ :

$$\begin{aligned} H(q)\dot{s} &= -c(q, \dot{q})s - Y\alpha(\hat{\phi} + w) + Y\alpha\phi - K_v s + F(\dot{q}) - F(\dot{q}) \\ &= -K_v s - c(q, \dot{q})s + Y\alpha(\phi - \hat{\phi} - w) \\ &= -K_v s - c(q, \dot{q})s + Y\alpha(\tilde{\phi} - w) \end{aligned} \quad (5.2-13)$$

Por lo tanto se propone una función de Lyapunov:

$$V = \frac{1}{2} s^T H(q) s$$

derivando

$$\dot{V} = s^T \dot{H}(q) s + \frac{1}{2} s^T \dot{H}(q) s$$

sustituyendo (5.2-13) en la ecuación anterior:

Instituto de Ingeniería  
Universidad Veracruzana

$$\begin{aligned} \dot{V} &= \frac{1}{2} s^T \dot{H}(q)s + s^T [-K_v s - c(q, \dot{q})s + Ya(\tilde{\varphi} - w)] \\ &= \frac{1}{2} s^T [\dot{H}(q) - 2c(q, \dot{q})]s + s^T [Ya(\tilde{\varphi} - w)] - s^T K_v s \\ &= -s^T K_v s + s^T Ya(\tilde{\varphi} - w) \\ &\leq \|s^T\| \|Ya\| \|\tilde{\varphi}\| \\ &\leq \rho \end{aligned}$$

$$\leq -s^T K_v s + \rho \underbrace{\|Ya^T s\| - w^T Ya s}_{w^T Ya^T s = \rho \|Ya^T s\|}$$

$$w^T Ya^T s = \rho \|Ya^T s\|$$

$$w^T = \rho \frac{Ya^T s}{\|Ya^T s\|}$$

Por lo tanto sea  $q_i$  una señal continua acotada con por lo menos primera y segunda derivadas continuas y sea  $\varepsilon > 0$ . Entonces  $w$  está dada por:

$$w = \begin{cases} \rho \frac{Ya^T s}{\|Ya^T s\|} & \text{si } \|Ya^T s\| > \varepsilon \\ \frac{\rho}{\varepsilon} Ya^T s & \text{si } \|Ya^T s\| \leq \varepsilon \end{cases}$$

con  $\|\tilde{\varphi}\| \leq -\rho$

Entonces la ley de control (5.2-12) es continua y el sistema en lazo cerrado (5.2-13) es uniforme finalmente acotado (aunque no es cero).

5.2.6.2 Análisis de estabilidad.

Sea:

$$V = \frac{1}{2} s^T H(q) s + e^T \Lambda K_V e$$

derivando:

$$\dot{V} = \frac{1}{2} s^T \dot{H}(q) s + 2\dot{e}^T \Lambda K_V e + s^T H(q) \dot{s}$$

sustituyendo (5.2-13):

$$\dot{V} = \frac{1}{2} s^T \dot{H}(q) s + 2\dot{e}^T \Lambda K_V e + s^T [-c(q, \dot{q}) s - K_V s + Y a(\tilde{\varphi} + w)]$$

Para esto se puede notar que:

$$\begin{aligned} s^T K_V s &= (\dot{e}^T + e^T \Lambda) K_V (\dot{e} + \Lambda e) \\ &= \dot{e}^T K_V \dot{e} + \dot{e}^T K_V \Lambda e + e^T \Lambda K_V \dot{e} + e^T \Lambda K_V \Lambda e \\ &= \dot{e}^T K_V \dot{e} + 2\dot{e}^T \Lambda K_V e + e^T \Lambda K_V \Lambda e \end{aligned}$$

Por lo tanto se tiene:

$$\begin{aligned} \dot{V} &= \frac{1}{2} s^T [\dot{H}(q) - 2c(q, \dot{q})] s + 2\dot{e}^T \Lambda K_V e - \dot{e}^T K_V \dot{e} - 2\dot{e}^T \Lambda K_V e - e^T \Lambda K_V \Lambda e + s^T Y a(\tilde{\varphi} + w) \\ &= -e^T \Lambda K_V \Lambda e - \dot{e}^T K_V \dot{e} + s^T Y a(\tilde{\varphi} + w) \end{aligned}$$

Simplificando:

$$\dot{V} \cong -x^T Q x + s^T Y a(\tilde{\varphi} + w)$$

con

$$x = \begin{bmatrix} e \\ \dot{e} \end{bmatrix} ; \quad Q = \begin{bmatrix} \Lambda K_V \Lambda & 0 \\ 0 & K_V \end{bmatrix}$$

Para el caso de  $w$  se distinguen dos casos:

Primero. Cuando  $\|Ya^T s\| > \varepsilon$

Entonces:  $w = \rho \frac{Ya^T s}{\|Ya^T s\|}$

Sustituyendo en  $\dot{V}$

$$\dot{V} = -x^T Qx + s^T Ya \left( \tilde{\varphi} + \frac{\rho Ya^T s}{\|Ya^T s\|} \right)$$

$$\dot{V} \leq -x^T Qx + \|\tilde{\varphi}\| s^T Ya + \frac{\rho s^T Ya Ya^T s}{\|Ya^T s\|}$$

Aplicando propiedades de normas:

$$\begin{aligned} \dot{V} &= -x^T Qx + \|\tilde{\varphi}\| \|Ya^T s\| + \frac{\rho \|Ya^T s\|^2}{\|Ya^T s\|} \\ &\leq -x^T Qx - \rho \|Ya^T s\| + \rho \|Ya^T s\| \end{aligned}$$

$$\dot{V} = -x^T Qx$$

Segundo. Cuando

$$\|Ya^T s\| \leq \varepsilon$$

$$w = \frac{\rho}{\varepsilon} Ya^T s$$

Sustituyendo en  $\dot{V}$ :

$$\dot{V} = -x^T Qx + s^T Ya \left( \tilde{\varphi} + \frac{\rho}{\varepsilon} Ya^T s \right)$$

$$= -x^T Qx + \|\tilde{\varphi}\| \|Ya^T s\| + \frac{\rho}{\varepsilon} \|Ya^T s\|^2$$

$$\leq -x^T Qx - \rho \|Ya^T s\| + \frac{\rho}{\varepsilon} \|Ya^T s\|^2$$

$$= -x^T Qx - \rho \left( \|Ya^T s\| + \frac{1}{\varepsilon} \|Ya^T s\|^2 \right)$$

Podemos observar que \* tiene la forma:

$$f(y) = y + \frac{y^2}{\varepsilon}$$

Para obtener el máximo se deriva:

$$\frac{df(y)}{dy} = 1 + \frac{2y}{\varepsilon} = 0$$

$$\Rightarrow y = -\frac{\varepsilon}{2}$$

sustituyendo y tenemos:

$$f\left(-\frac{\varepsilon}{2}\right) = -\frac{\varepsilon}{2} + \frac{\varepsilon^2}{4\varepsilon} = -\frac{\varepsilon}{2} + \frac{\varepsilon}{4} = -\frac{\varepsilon}{4}$$

Por lo tanto, independientemente del valor  $\|Ya^T s\|$  se tiene, sustituyendo en  $\dot{V}$ :

$$\dot{V}(x) \leq -x^T Qx + \frac{\varepsilon}{4} \rho$$

con  $Q > 0$

Δ

## 4.2.7 Control Robusto de Modo Deslizante (1ª Forma).

Considerando algunas definiciones antes mencionadas:

$$\begin{aligned} e &= q_d - q \\ \dot{q}_r &= \dot{q}_d + \Lambda e \\ s &= \dot{q}_r - \dot{q} = \dot{q}_d + \Lambda e - \dot{q} = \dot{e} + \Lambda e \\ &\text{con } \Lambda > 0 \end{aligned}$$

derivando

$$\therefore \dot{s} = \ddot{e} + \Lambda \dot{e}$$

(4.2-14)

Despejando  $\ddot{q}$  de la ecuación (4.1-2):

$$\ddot{q} = H^{-1}(q)(\tau - N)$$

(4.2-15)

donde:  $N = c(q, \dot{q})\dot{q} + F(\dot{q}) + g(q)$

sustituyendo (4.2-15) en (4.2-14):

$$\dot{s} = [-H^{-1}(q)(\tau - N) + \dot{q}_d] + \Lambda \dot{e}$$

$$\dot{s} = -H^{-1}(q)(\tau - N) + \Lambda \dot{e} + \dot{q}_d$$

$$\dot{s} = -H^{-1}(q)(\tau - N) + \alpha(q, q_d)$$

(4.2-16)

donde:  $\alpha(q, q_d) = \Lambda \dot{e} + \dot{q}_d \equiv \ddot{q}_r$

o

$$H(q)\dot{s} = -(\tau - N) + H(q)\alpha(q, q_d)$$

(4.2-17)

Ahora se propone una función de Lyapunov:

$$V = \frac{1}{2} s^T I s$$

donde:  $I$  = Matriz identidad  $\in R^{n \times n}$  ;  $n$  = grados de libertad (GDL)

derivando:

sustituyendo (4.2-16)

$$\dot{V} = s^T [-H^{-1}(q)(\tau - N) - \alpha(q, \dot{q}_d)]$$

$$\dot{V} = -Kq \text{sign}(s)$$

Por LaSalle:

$$\text{Si } Kq > 0 \text{ y } e \rightarrow 0, s \rightarrow 0, V = 0$$

Por lo tanto  $\tau$  quedaría:

$$-H^{-1}(q)(\tau - N) + \alpha(q, \dot{q}_d) = -Kq \text{sign}(s)$$

$$H^{-1}(q)(\tau - N) = [Kq \text{sign}(s) + \alpha(q, \dot{q}_d)]$$

$$\tau - N = H(q)[Kq \text{sign}(s) + \alpha(q, \dot{q}_d)]$$

∴

$$\tau = H(q)[Kq \text{sign}(s) + \alpha(q, \dot{q}_d)] + N$$

NOTA: este control se puede mejorar con la función saturación [3] ó  $\tanh(s)$ [7].

## 4.2.8 Control Robusto de Modo Deslizante (2ª Forma).

Tomando las definiciones anteriores se puede proponer otra función de Lyapunov:

$$V = \frac{1}{2} s^T H(q) s$$

derivando:

$$\dot{V} = \frac{1}{2} s^T \dot{H}(q) s + s^T H(q) \dot{s}$$

sustituyendo (4.2-17):

$$\dot{V} = \frac{1}{2} s^T \dot{H}(q) s + s^T [-\tau + c(q, \dot{q})(\dot{q}_r - s) + F(\dot{q}) + g(q) + H(q) \alpha(q, q_d)]$$

$$\dot{V} = \frac{1}{2} s^T [\dot{H}(q) - 2c(q, \dot{q})] s + s^T [-\tau + c(q, \dot{q})\dot{q}_r + F(\dot{q}) + g(q) + H(q) \alpha(q, q_d)]$$

$$\dot{V} = -Kq|s| \leq 0$$

por LaSalle:

$$e \rightarrow 0, s \rightarrow 0, \dot{V} \rightarrow 0.$$

Por lo tanto  $\tau$  quedaría:

$$\tau = Kq \operatorname{sign}(s) + H(q) \alpha(q, q_d) + \beta(q, \dot{q})$$

donde:  $\beta(q, \dot{q}) = c(q, \dot{q})\dot{q}_r + F(\dot{q}) + g(q)$

NOTA: este control se puede mejorar con la función saturación [3] ó  $\tanh(s)$ [7].

## V. SIMULACIÓN Y RESULTADOS

Para efectos de simulación se utiliza un programa muy poderoso en la actualidad Matlab, el cual se basa en el uso de operaciones matriciales, pero no solo eso si no que también abarca cuestiones matemáticas como: funciones trigonométricas, funciones especiales u originadas por el usuario, comandos que simulan operaciones de control clásico y moderno, etc.

### 5.1. El programa MATLAB

MATLAB es el nombre abreviado de "MATrix LABoratory". MATLAB es un programa para realizar cálculos numéricos con *vectores* y *matrices*. Como caso particular puede también trabajar con números escalares, tanto reales como complejos. Una de las capacidades más atractivas es la de realizar una amplia variedad de *gráficos* en dos y tres dimensiones. MATLAB tiene también un lenguaje de programación propio. Las simulaciones se hacen en la versión 5.3 de este programa, aparecida a comienzos de 1999. MATLAB se puede arrancar como cualquier otra aplicación de *Windows 95/98/NT* como se muestra en el apéndice B.

En la ventana inicial (apéndice B) se sugieren ya algunos comandos para el usuario inexperto que quiere echar un vistazo a la aplicación. En dicha ventana aparece también el *prompt* característico de MATLAB (»>). Esto quiere decir que el programa está preparado para recibir instrucciones. Puede hacerse que aparezca un saludo inicial personalizado por medio de un *fichero de comandos* personal que se ejecuta cada vez que se entra en el programa (el fichero *startup.m*, que debe estar en un directorio determinado, por ejemplo *C:\Matlab*, o *G:\Matlab*).

## Ejemplo 5.1-1. Utilizando Matlab.

(a)

Para apreciar desde el principio la potencia de MATLAB, se puede comenzar por escribir la siguiente línea, a continuación del *prompt*. Al final hay que pulsar *intro*.

```
» A=rand(6) , B=inv(A) , B*A
```

```
A =
0.9501 0.4565 0.9218 0.4103 0.1389 0.0153
0.2311 0.0185 0.7382 0.8936 0.2028 0.7468
0.6068 0.8214 0.1763 0.0579 0.1987 0.4451
0.4860 0.4447 0.4057 0.3529 0.6038 0.9318
0.8913 0.6154 0.9355 0.8132 0.2722 0.4660
0.7621 0.7919 0.9169 0.0099 0.1988 0.4186

B =
5.7430 2.7510 3.6505 -0.1513 -6.2170 -2.4143
-4.4170 -2.5266 -1.4681 -0.5742 5.3399 1.5631
-1.3917 -0.6076 -2.1058 -0.0857 1.5345 1.8561
-1.6896 -0.7576 -0.6076 -0.3681 3.1251 -0.6001
-3.6417 -4.6087 -4.7057 2.5299 6.1284 0.9044
2.7183 3.3088 2.9929 -0.1943 -5.1286 -0.6537

ans =
1.0000 0.0000 0 0.0000 0.0000 -0.0000
0.0000 1.0000 0.0000 0.0000 -0.0000 0.0000
0 0 1.0000 -0.0000 -0.0000 0.0000
0.0000 0 -0.0000 1.0000 -0.0000 0.0000
-0.0000 0.0000 -0.0000 -0.0000 1.0000 0.0000
-0.0000 -0.0000 -0.0000 -0.0000 -0.0000 1.0000
```

En realidad, en la línea de comandos anterior se han escrito tres instrucciones diferentes, separadas por comas. Como consecuencia, la respuesta del programa tiene tres partes también, cada una de ellas correspondiente a una de las instrucciones. Con la primera instrucción se define una matriz cuadrada (6x6) llamada **A**, cuyos elementos son números aleatorios entre cero y uno (aunque aparezcan sólo 4 cifras, han sido

calculados con 16 cifras). En la segunda instrucción se define una matriz **B** que es igual a la inversa de **A**. Finalmente se ha multiplicado **B** por **A**, y se comprueba que el resultado es la matriz identidad<sup>1</sup>.

Otro de los puntos fuertes de MATLAB son los gráficos, que se verán con más detalle en una sección posterior.

(b)

Se puede teclear la siguiente línea y pulsar *intro*:

```
» x=-4:.01:4; y=sin(x); plot(x,y), grid, title('Función seno(x)');
```

Se formará una nueva ventana en la que aparece representada la función *sin(x)*. Esta figura tiene un título "Función seno(x)" y una cuadrícula o "grid". En realidad la línea anterior contiene también varias instrucciones separadas por comas o puntos y comas. En la primera se crea un vector **x** con valores entre -4 y 4 separados por una centésima. A continuación se crea un vector **y**, cada uno de cuyos elementos es el seno del correspondiente elemento del vector **x**. Después se dibujan los valores de **y** en ordenadas frente a los de **x** en abscisas. Las dos últimas instrucciones establecen la cuadrícula y el título. Un pequeño aviso antes de seguir adelante. Es posible recuperar comandos anteriores de Matlab y moverse por dichos comandos con las teclas-flechas  $\uparrow$  y  $\downarrow$ . Al pulsar la primera de dichas flechas aparecerá el comando que se había introducido inmediatamente antes.

<sup>1</sup> Al invertir la matriz y al hacer el producto posterior se han introducido pequeños errores numéricos de redondeo en el resultado, lo cual hace que no todos los elementos cero del resultado aparezcan de la misma forma.

De modo análogo es posible moverse sobre la línea de comandos con las teclas  $\leftarrow$  y  $\rightarrow$ , ir al principio de la línea con la tecla *Inicio*, al final de la línea con *Fin*, y borrar toda la línea con *Esc*. Para borrar todas las salidas anteriores de MATLAB y dejar limpia la ventana principal se pueden utilizar las funciones *clc* y *home*. Si se desea salir del programa, basta teclear los comandos *quit* o *exit*, o bien elegir *Exit* MATLAB en el menú *File* (también se puede utilizar el *Alt+F4* de todas las aplicaciones de Windows).

### 5.2. Simulación en lazo abierto.

Una vez obtenido el modelo del robot manipulador, se simula en lazo abierto a entrada cero en un paquete llamado SIMULINK [6] de Matlab para Windows [2], en forma de bloques, como se muestra en la figura 5.1.

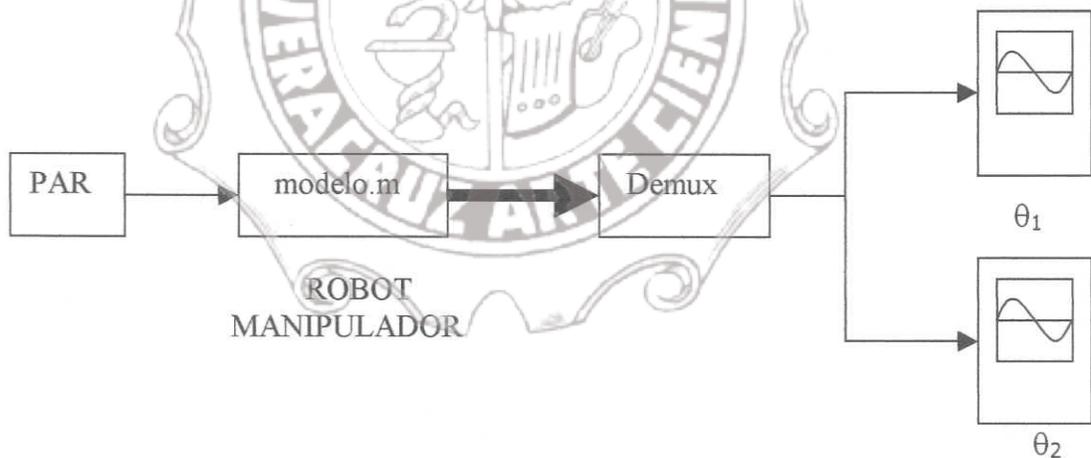


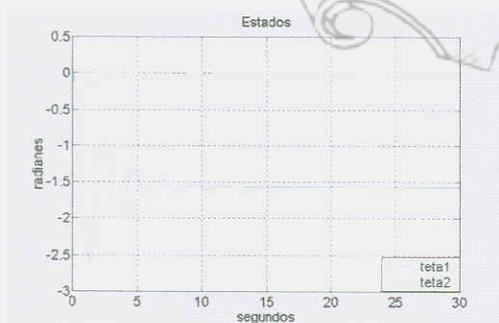
Figura 5.1 Simulación lazo abierto.

En la figura 5.1 podemos observar que el bloque PAR es la entrada (para este caso es cero) de ROBOT MANIPULADOR el cual está formado por el archivo

modelo.m, este a su vez es un programa de Matlab y es llamado por un bloque *Matlab Function*. La línea gruesa que se muestra a la entrada del Demultiplexor nos indica los estados están en forma de vectores, donde sus elementos se identifican con las etiquetas de  $\theta_1$  y  $\theta_2$  (para más detalles ver apéndice B).

En las figuras 5.2 – 5.7, podemos observar estas salidas, las cuales más adelante serán dirigidas por los diferentes controladores a un punto determinado. También se muestran diferentes gráficas en donde las condiciones iniciales son iguales y diferentes de cero, y observaremos como actúa la fricción sobre el sistema a controlar estabilizándolo en un determinado valor ( $-\pi/2$ ), esto nos demuestra que el modelo en lazo abierto es correcto.

Podemos observar como la primera función  $\theta_2$  comienza en un valor dado (condición inicial) y estabiliza en cero, esto es porque la segunda articulación fue considerada con respecto a la primera con cero radianes. La segunda función  $\theta_1$  inicia igualmente en un valor dado (condición inicial) y estabiliza en  $-\pi/2$  ( $-1.57079$  rad.), Es decir, se considera la posición de las articulaciones diferentes de cero y retornan a su punto de equilibrio o de reposo [3], respectivamente.



**Figura 5.2.** Simulación en lazo abierto con condiciones iniciales iguales a cero.



**Figura 5.3.** Simulación en lazo abierto con condiciones iniciales  $\theta_1 = \pi/2$  y  $\theta_2 = -\pi/2$ .



Figura 5.4. Simulación en lazo abierto con condiciones iniciales  $\theta_1 = 0$  y  $\theta_2 = \pi/2$ .

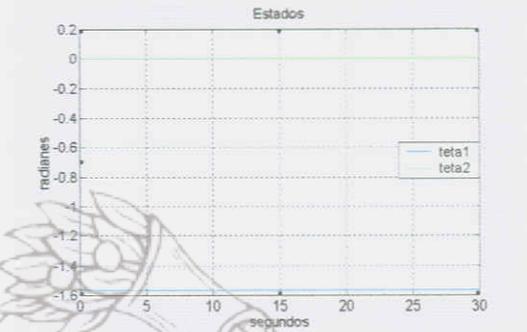


Figura 5.5. Simulación en lazo abierto con condiciones iniciales  $\theta_1 = -\pi/2$  y  $\theta_2 = 0$ .

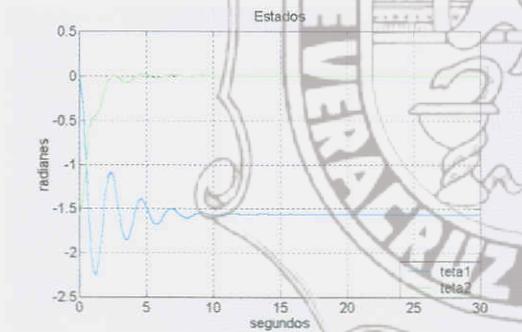


Figura 5.6. Simulación en lazo abierto con condiciones iniciales  $\theta_1 = 0$  y  $\theta_2 = -\pi/2$ .

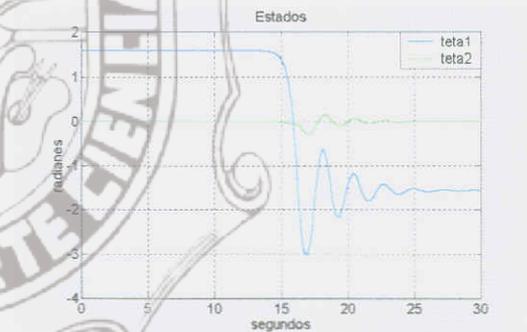


Figura 5.7. Simulación en lazo abierto con condiciones iniciales  $\theta_1 = \pi/2$  y  $\theta_2 = 0$ .

Las subrutinas utilizadas para esta simulación se muestran en el apéndice A, a continuación se muestra el programa para simular el modelo en Matlab (figura 5.8), este se utiliza en lazo abierto y en lazo cerrado.

```

funcion salida=modelo2(u) %esta función es un modelo para simular el robot de 2 DOF.
%----- Datos Fijos -----
A=[0,0,1,0;0,0,0,1;0,0,0,0;0,0,0,0];
B=[0,0;0,0;1,0;0,1];

%----- Datos a la Escala por los datos del robot -----
ini;
%----- Datos de los grados de libertad -----
tetal= u(1);
teta2= u(2);
dtetal=u(3);
dteta2=u(4);

par1=u(5);
par2=u(6);
tetad1=u(7);
tetad2=u(8);

par=[par1;par2];

x=[u(1);u(2);u(3);u(4)];

%----- Datos de los miembros -----
m11=((m1+m2)*(a1^2)+(m2*(az^2)+(2*m2*a1*a2*cos(teta2)));
m12=(m2*(a2^2)+(m2*a1*a2*cos(teta2)));
m21=m12;
m22=m2*(a2^2);

M=[m11,m12;m21,m22];
MINV=pinv(M);

c1=-m2*a1*a2*(2*dtetal*dteta2+(dteta2^2))*sin(teta2);
c2=m2*a1*a2*(dtetal^2)*sin(teta2);

C=[c1;c2];

g1=((m1+m2)*g*a1*cos(tetal)+(m2*g*a2*cos(tetal-teta2));
g2=m2*g*a2*cos(tetal+teta2);

G=[g1;g2];

f1=(vis1*dtetal)+(k1*tanh(dtetal));
f2=(vis2*dteta2)+(k2*tanh(dteta2));

F=[f1;f2];

N=C+G+F;

UU=MINV*(par-N);

%----- Datos de los grados de libertad -----
sys= A*x + B*UU;

%----- Datos de los grados de libertad -----
salida=[sys(1,1) sys(2,1) sys(3,1) sys(4,1) tetad1 tetad2];
    
```

Figura 5.8. Subrutina para simular el modelo del manipulador usando Matlab.

### 5.3. Simulación en lazo cerrado.

Para la simulación en lazo cerrado se pretende utilizar una forma estándar (apéndice B), para poder seleccionar los diferentes controladores diseñados, quedando de la siguiente forma:

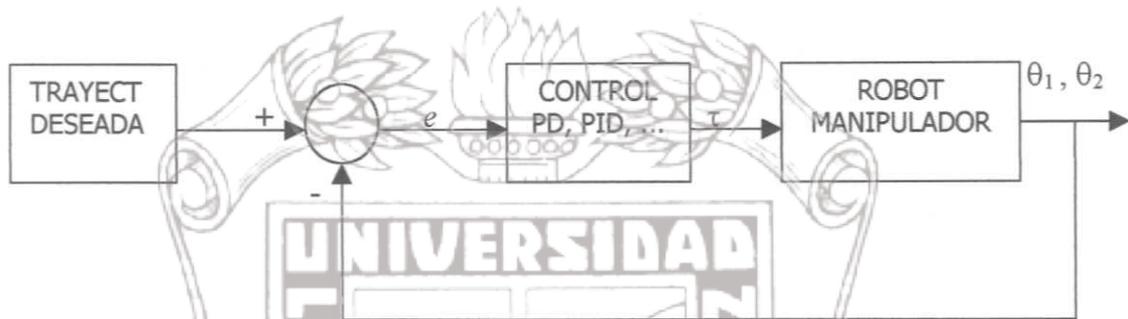


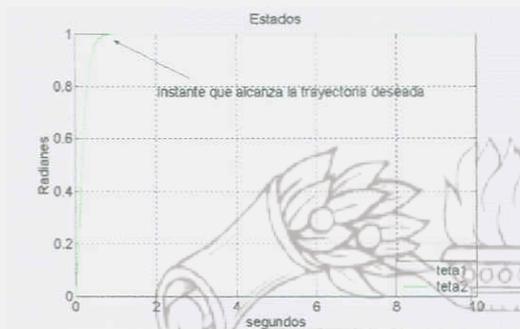
Figura 5.9. Diagrama esquemático para controlar el robot manipulador.

En donde, las trayectorias generadas son retroalimentadas y restadas con las deseadas hasta que el error es lo más cercano a cero. En este caso,  $\tau$  es la señal de control enviada a la planta<sup>2</sup> como el torque generado. Y  $e$  es error ya sea expresado en su posición, velocidad o aceleración. Se consideran dos tipos de controles: *regulador* y *seguidor*. Esto es por una simple causa, cuando un control tiene como trayectoria deseada una constante se le llama regulador, cuando es variante en el tiempo se le llama seguidor. A continuación se muestran unas gráficas obtenidas en la simulación en lazo cerrado, utilizando los controladores diseñados en la sección 4.2 como reguladores, y comparándolas entre sí.

<sup>2</sup>En este caso llamamos 'Planta' a el modelo del manipulador, y en general al sistema que se quiera controlar. Para más detalles ver Kuo Benjamín C. *Sistemas Automáticos de Control*. Ed. CECSA 1991. Segunda Edición. Cap.1.

## 5.3.1 Regulación.

### 5.3.1.1 Control PD par-calculado y PD (por Lyapunov).

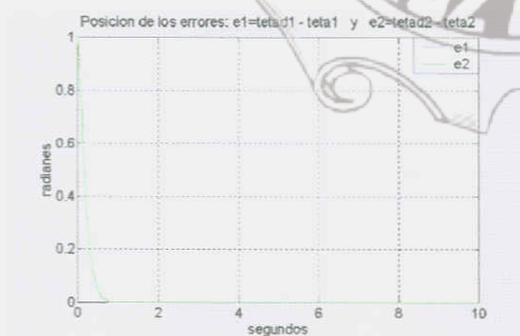


**Figura 5.10.** Ángulos de las articulaciones  $\theta_1$  y  $\theta_2$  (rad) para el control PD-par-calculado, con un  $k_p = 100I$ ,  $k_v = 20I$ .



**Figura 5.11.** Ángulos de las articulaciones  $\theta_1$  y  $\theta_2$  (rad) para el control PD, con un  $k_p = 100I$ ,  $k_v = 20I$ . Este control varía porque está diseñado por Lyapunov.

En el caso de la figura 5.10, no se puede apreciar  $\theta_1$  ya que tiene una regulación (seguimiento para trayectorias no variables en el tiempo) exacto con  $\theta_2$ , debido a la cancelación de las no linealidades por *linealización exacta* [3], aplicando retroalimentación de estados. Sus condiciones iniciales son iguales con cero.



**Figura 5.12.** Las posiciones de los errores tienden a cero, para el control PD-par-calculado, con un  $k_p = 100I$ ,  $k_v = 20I$ .



**Figura 5.13.** Se puede observar las posiciones de los errores que tienden a cero, para el control PD, con un  $k_p = 100I$ ,  $k_v = 20I$ .



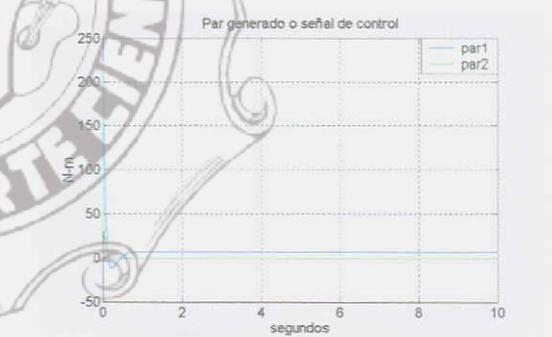
**Figura 5.14.** Se observa las velocidades de cambio de los errores que tienden a cero, para el control PD-par-calculado, con un  $k_p = 100I$ ,  $k_v = 20I$ .



**Figura 5.15.** Se observa las velocidades de cambio de los errores que tienden a cero, para el control PD, con un  $k_p = 100I$ ,  $k_v = 20I$ .



**Figura 5.16.** Se puede observar el torque (N-m) generado por el control PD-par-calculado.

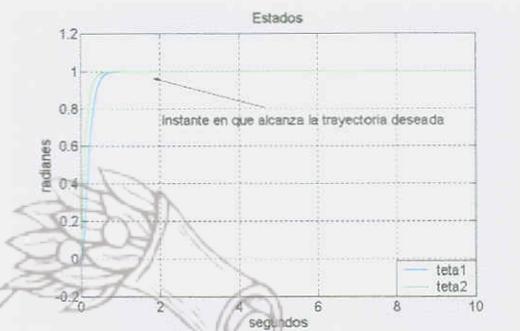


**Figura 5.17.** Se observa el torque (N-m) generado por el control PD.

## 5.3.1.2 Control PID y Estructura pasiva.



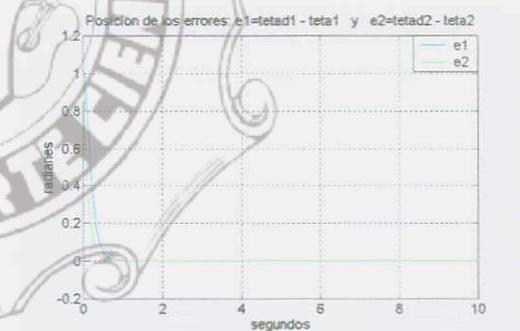
**Figura 5.18.** Ángulos de las articulaciones  $\theta_1$  y  $\theta_2$  (rad). Se puede observar el seguimiento de las trayectorias generadas con las trayectorias deseadas (escalón unitario) para el control PID, con un  $k_p = 100I$ ,  $k_v = 20I$  y  $k_i = 500I$ .



**Figura 5.19.** Ángulos de las articulaciones  $\theta_1$  y  $\theta_2$  (rad) para el control de ESTRUCTURA PASIVA.

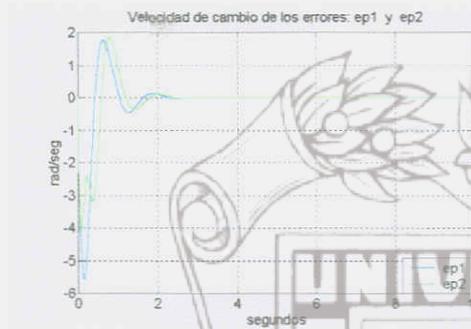


**Figura 5.20.** Se observa las posiciones de los errores que tienden a cero, para el control PID, con un  $k_p = 100I$ ,  $k_v = 20I$  y  $k_i = 500I$ .



**Figura 5.21.** Se puede observar las posiciones de los errores que tienden a cero, para el control de ESTRUCTURA PASIVA, con un  $k_e = 20I$ ,  $\lambda = 10I$ .

En la figura se puede observar el seguimiento de las trayectorias generadas con las trayectorias deseadas (escalón unitario) para el control de ESTRUCTURA PASIVA, con un  $ke = 20I$ ,  $\lambda = 10I$ .



**Figura 5.22.** Se observa las velocidades de cambio de los errores que tienden a cero, para el control PID, con un  $k_p = 100I$ ,  $k_v = 20I$  y  $k_i = 500I$ .



**Figura 5.23.** Se puede observar las velocidades de cambio de los errores que tienden a cero, para el control ESTRUCTURA PASIVA, con un  $ke = 20I$ ,  $\lambda = 10I$ .

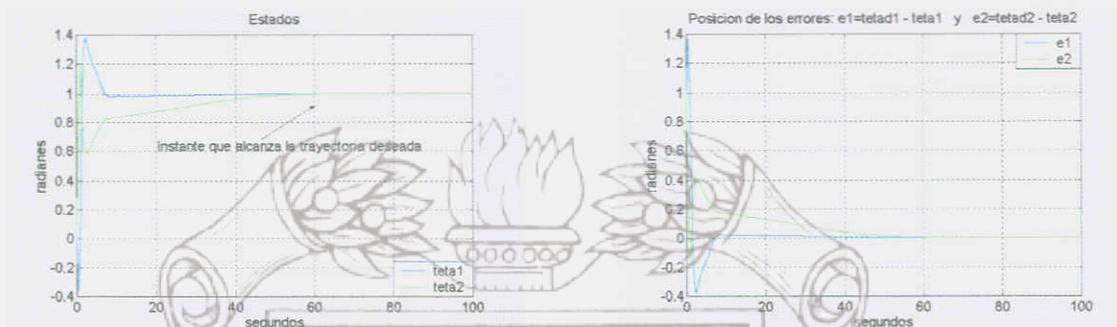


**Figura 5.24.** Se puede observar el torque (N-m) generado por el control PID.



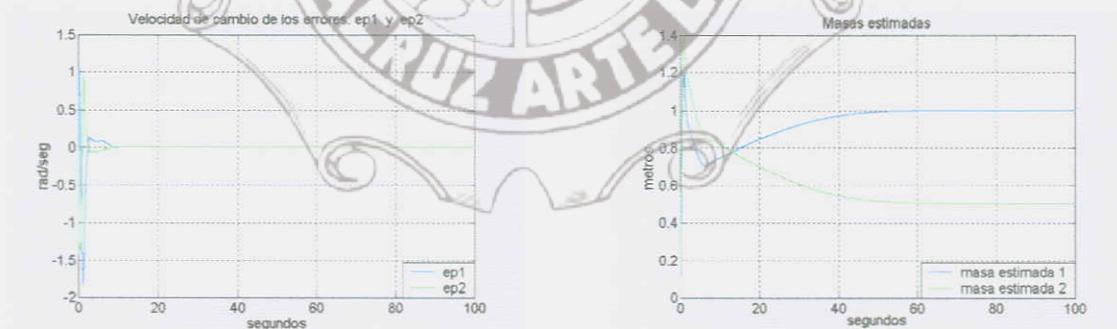
**Figura 5.25.** Se observa el torque (N-m) generado por el control de ESTRUCTURA PASIVA.

5.3.1.3 Control Adaptable. Se puede observar que debido a que sus funciones operativas son diferentes, no es posible compararlo con algún otro de la sección 4.2.



**Figura 5.26.** Ángulos de las articulaciones  $\theta_1$  y  $\theta_2$  (rad). Se puede observar el seguimiento de las trayectorias generadas, para el control ADAPTABLE, con un  $k = 1.5I$ ,  $\lambda = 1.5I$  y  $\gamma = (1/15)I$ .

**Figura 5.27.** Se puede observar las posiciones de los errores que tienden a cero, para el control ADAPTABLE.



**Figura 5.28.** Las velocidades de cambio de los errores tienden a cero, para el control ADAPTABLE.

**Figura 5.29.** Estimación de parámetros  $m_1=1$  y  $m_2=0.5$  del control ADAPTABLE.

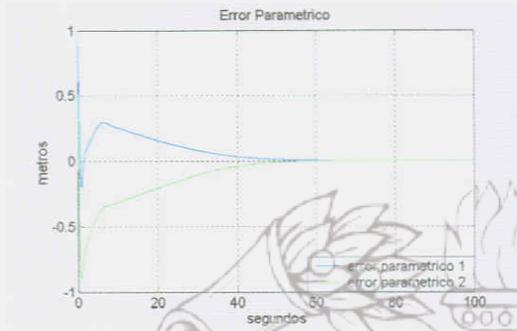


Figura 5.30. Errores paramétricos  $\tilde{\phi}_1$  y  $\tilde{\phi}_2$  del control ADAPTABLE.



Figura 5.31. Torque generado por el control ADAPTABLE.

### 5.3.1.4 Control Robusto compensado y de modo Deslizante.

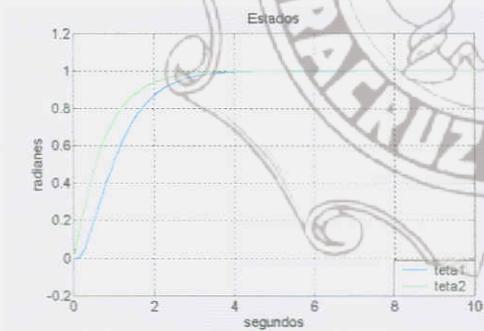


Figura 5.32. Ángulos de las articulaciones  $\theta_1$  y  $\theta_2$  (rad) del control ROBUSTO.

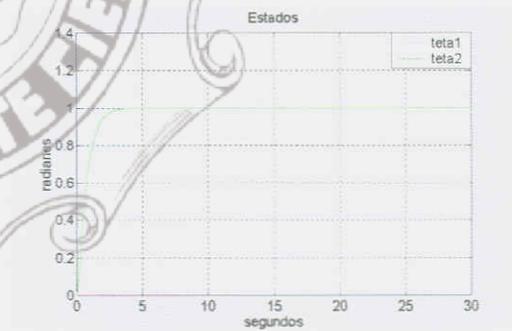
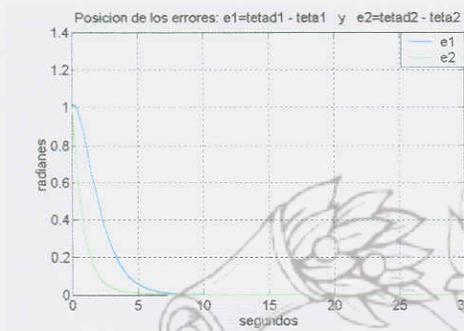


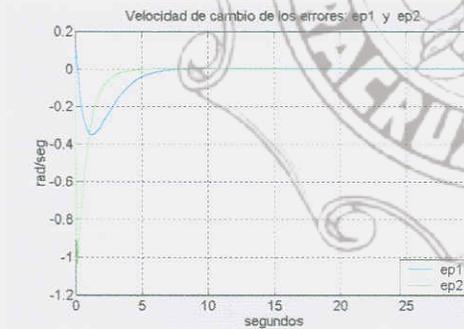
Figura 5.33. Ángulos de las articulaciones  $\theta_1$  y  $\theta_2$  (rad) para el control de modo DESLIZANTE.



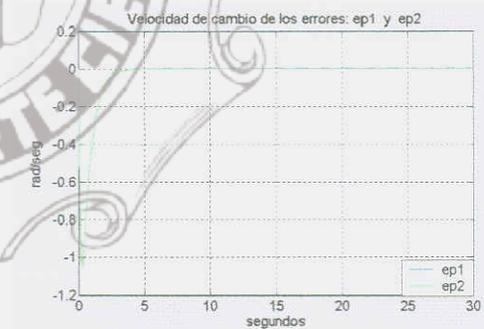
**Figura 5.34.** Se puede observar las posiciones de los errores que tienden a cero, para el control ROBUSTO.



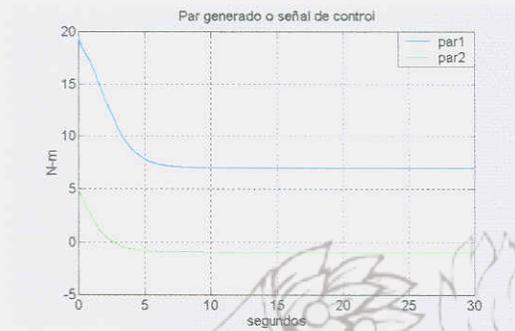
**Figura 5.35.** Las posiciones de los errores tienden a cero, para el control de modo DESLIZANTE.



**Figura 5.36.** Se puede observar las velocidades de cambio de los errores que tienden a cero, para el control ROBUSTO.



**Figura 5.37.** Se puede observar las velocidades de cambio de los errores que tienden a cero, para el control de modo DESLIZANTE.

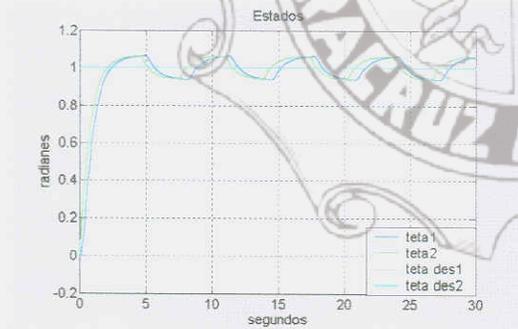


**Figura 5.38.** Torque generado por el control ROBUSTO.



**Figura 5.39.** Torque generado por el control de modo DESLIZANTE.

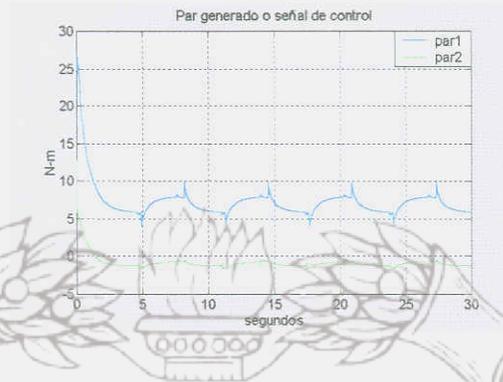
### 5.3.1.5 Deslizante (2ª opción)



**Figura 5.40.** Ángulos de las articulaciones  $\theta_1$  y  $\theta_2$  (rad) para el control de modo DESLIZANTE (2ª opción)



**Figura 5.41.** Los errores no tienden a cero, para el control de modo DESLIZANTE (2ª opción)



**Figura 5.42.** Torque generado por el control de modo DESLIZANTE (2ª opción)

Para las figuras 5.40-5.42, que es la segunda opción del control de modo deslizante, podemos decir que esta no cumple con los parámetros de diseño, ya que el error nunca es cero. Esto es muy importante de tomarse en cuenta al momento de diseñar un control, ya que no se puede proponer una función de Lyapunov al azar.

Caso distinto a las gráficas de los otros controles, donde se observa perfectamente como el error tiende a cero cuando el tiempo tiende a infinito.

5.3.2 Seguimiento.

5.3.2.1 Control PD par-calculado y PD (por Lyapunov).

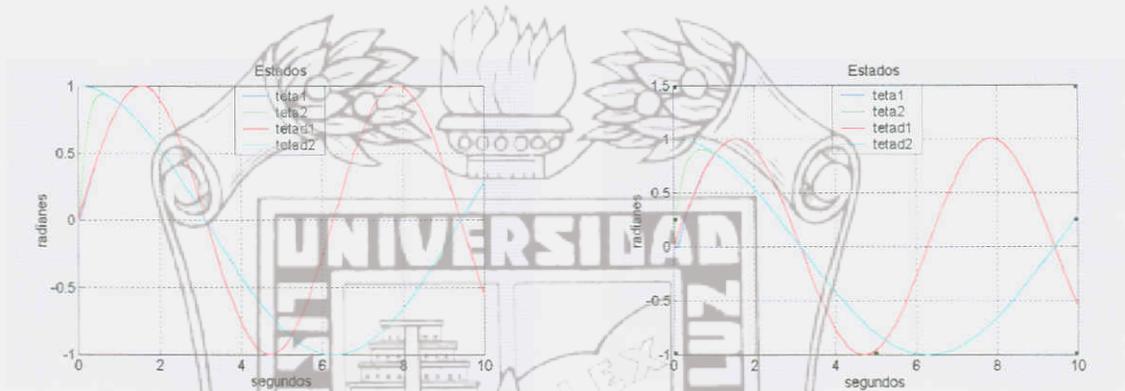


Figura 5.43. Ángulos de las articulaciones  $\theta_1$  y  $\theta_2$  (rad) para el control PD-par- calculado, con un  $k_p = 100I$ ,  $k_v = 20I$ .

Figura 5.44. Ángulos de las articulaciones  $\theta_1$  y  $\theta_2$  (rad) para el control PD, con un  $k_p = 100I$ ,  $k_v = 20I$ . Este control varía porque está diseñado por Lyapunov.

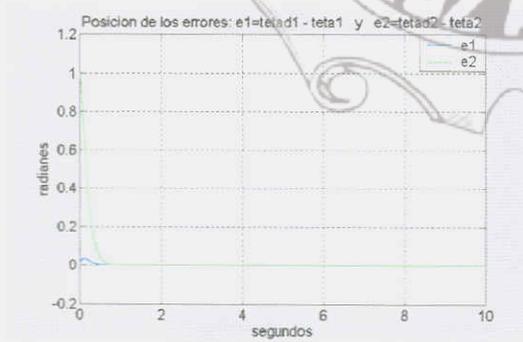


Figura 5.45. Los errores tienden a cero, para el control PD-par-calculado, con un  $k_p = 100I$ ,  $k_v = 20I$ .



Figura 5.46. Se puede observar las posiciones de los errores que tienden a cero, para el control PD, con un  $k_p = 100I$ ,  $k_v = 20I$ .

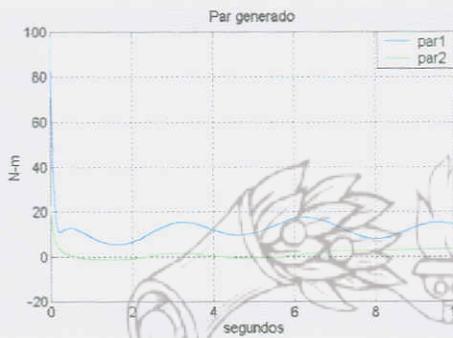


Figura 5.47. Torque (N-m) generado por el control PD-par-calculado.

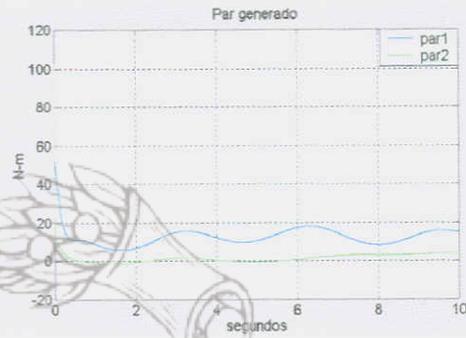


Figura 5.48. Torque (N-m) generado por el control PD.

### 5.3.2.2 Control PID par-calculado y Estructura Pasiva.

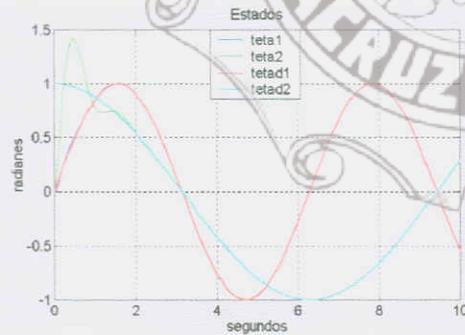


Figura 5.49. Ángulos de las articulaciones  $\theta_1$  y  $\theta_2$  (rad). Se puede observar el seguimiento de las trayectorias generadas con las trayectorias deseadas (escalón unitario) para el control PID, con un  $k_p = 100I$ ,  $k_v = 20I$  y  $k_i = 500I$ .

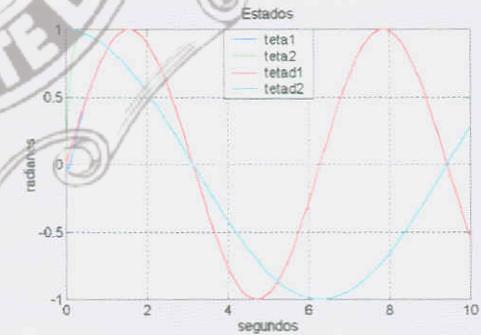


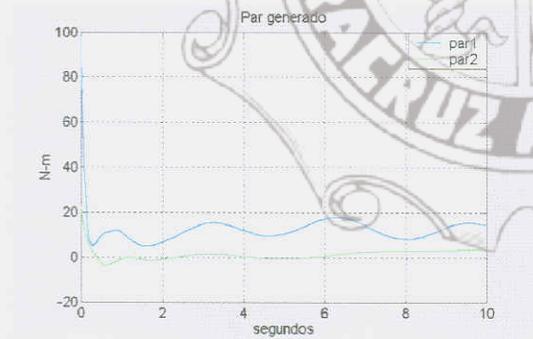
Figura 5.50. Ángulos de las articulaciones  $\theta_1$  y  $\theta_2$  (rad) para el control de ESTRUCTURA PASIVA.



**Figura 5.51.** Se observa las posiciones de los errores que tienden a cero, para el control PID, con un  $k_p = 100I$ ,  $k_v = 20I$  y  $k_i = 500I$ .



**Figura 5.52.** Posiciones de los errores que tienden a cero, para el control de ESTRUCTURA PASIVA, con un  $k_e = 20I$ ,  $\lambda = 10I$ .



**Figura 5.53.** Torque (N-m) generado por el control PID.



**Figura 5.54.** Se observa el torque (N-m) generado por el control ESTRUCTURA PASIVA.

5.3.2.3 Adaptable. En este caso se reduce el tiempo de integración para que se pueda apreciar el seguimiento.

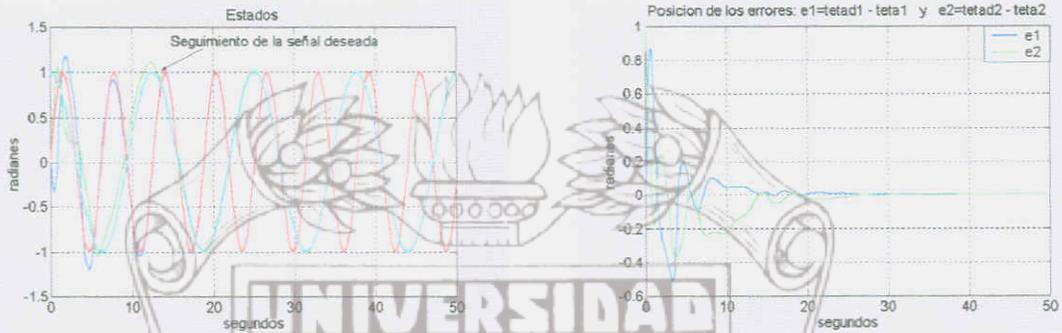


Figura 5.55. Seguimiento de las trayectorias deseadas.

Figura 5.56. Posición de los errores.



Figura 5.57. Estimación de parámetros (masas  $m_1$  y  $m_2$ ).

Figura 5.58. Par generado o señal de control.

## 5.3.2.4 Control Robusto compensado y de modo Deslizante.

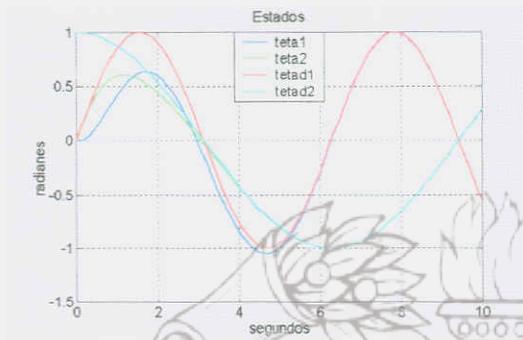


Figura 5.59. Seguimiento de las trayectorias deseadas. Control Robusto.

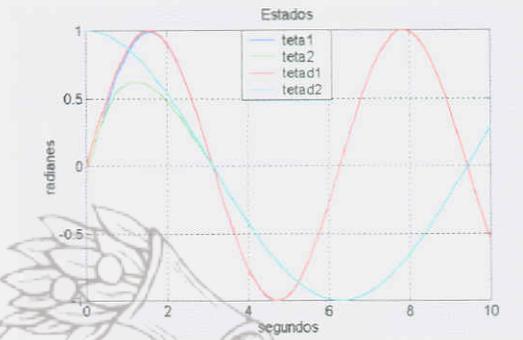


Figura 5.60. Seguimiento de las trayectorias deseadas. Control Deslizante.



Figura 5.61. Errores de posición. Control robusto.

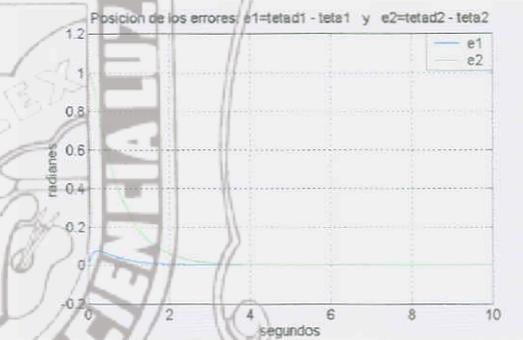


Figura 5.62. Errores de posición. Control Deslizante.

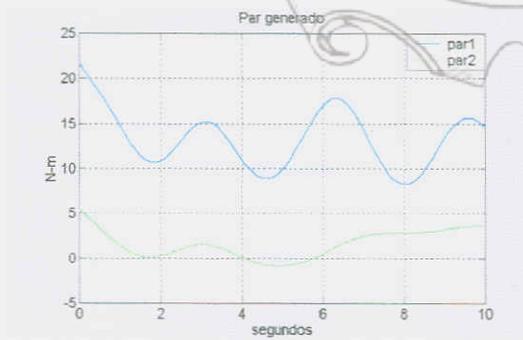


Figura 5.63. Par generado por el control Robusto.

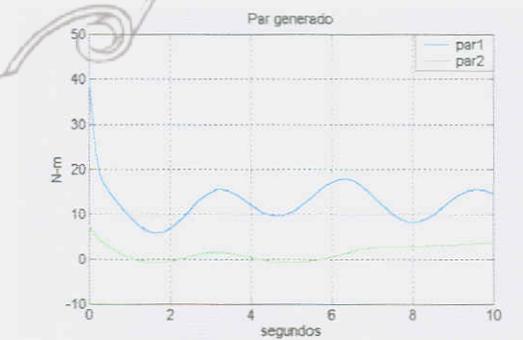


Figura 5.64. Par generado por el control Deslizante.

## CONCLUSIÓN

Los controladores diseñados en esta tesis, fueron basados en diferentes técnicas de control, *pasividad, Lyapunov, linealización*; todo esto dentro del control moderno para sistemas no lineales. El seguimiento es muy bueno, aunque se diferencian en la señal de control generada que se envía al Robot; podemos observar como los controles robustos son muy buenos ya que consideran variación de parámetros y algunos se muestran inmunes a las perturbaciones. Todo esto se observa en las gráficas de la señal de control generada o el par, su demanda de par de arranque es poca lo que ayuda a nuestro actuador a no forzarse tanto.

Es muy importante considerar este aspecto, ya que aunque observemos que nuestro control diseñado esté operando correctamente, la señal de control puede ocasionar problemas a nuestros sistemas de actuadores. Este problema lo presentan los controles PD y PID pero se puede solucionar utilizando un buen actuador.

Por otro lado, Matlab, es una herramienta muy poderosa que se utiliza en la actualidad para simular sistemas de control, con esto se puede observar y analizar el comportamiento de los mismos. Además, un paquete de simulación utilizado en este programa es el SIMULINK, el cual es mucho más práctico y fácil de usar, ya que se basa en el uso de funciones de bloques; por lo que esta tesis anexa un programa realizado en Matlab-SIMULINK, de los controladores diseñados, en forma estándar para propósitos didácticos.

## BIBLIOGRAFIA

1. Spong, M. W., and M. Vidyasagar, *Robot Dynamics and Control*. Ed. Wiley 1989
2. García y Rodríguez, *Aprenda Matlab 5.3 como si estuviera en primero* ESISS 1999
3. Khalil H. K. *Nonlinear Systems*. Ed. Prentice Hall 1996
4. Barrientos, A., *Fundamentos de Robótica* Ed. Mc Graw Hill 1996
5. Groover, M. P., *Robótica Industrial* Ed. Mc Graw Hill 1990
6. *Simulink User's Guide* The Math Works, Inc. 1992
7. Lewis, F. L., *Control of Robot Manipulators* Ed. Macmillan 1993
8. Tippens, P. E., *FISICA, Conceptos y aplicaciones* Ed. Mc Graw Hill 1974
9. Zill, D. G., *Ecuaciones Diferenciales* Ed. Iberoamérica 1988
10. Kuo, B. C., *Sistemas Automáticos de Control* Ed. CECSA 1994
11. Schilling, D. L., *Circuitos Electrónicos Discretos e Integrados* Ed. Mc Graw Hill 1990
12. Mckerrow, P. J., *Introduction to Robotics* Addison-wesley publishing Company 1991
13. Coiffet, P. y M. Chirouze, *Elementos de Robótica* Ed. Gustavo Gili, S.A. 1986
14. Marion, J. B., *Classical Dynamics*. New York: Academic Press, 1965
15. Åmström, K. J., and B. Wittenmark, *Computer-Controlled Systems: Theory and Desing*. Ed. Prentice Hall 1990

## APÉNDICE A

A continuación se muestran las subrutinas utilizadas para realizar las simulaciones hechas en Matlab.

```

%----- INI -----
%----- CONSTANTES -----
a1 =1;
a2 =0.5;
m1 =1.0;
m2 =0.5;
vis1=1;
vis2=1;
K1 =1;
K2 =1;

g = 9.81;

matiden=[1,0;0,1];

kp1=100;
kp2=100;
kv1=20;
kv2=20;
kk1=0.5;
kk2=0.5;

Kp= [kp1,0;0,kp2];
Kv= [kv1,0;0,kv2];
Ki= [kk1,0;0,kk2];

lmbd=10;
LAMBDA=lmbd*matiden;
Ke1=10;
Ke2=5;
Ke=[Ke1,0;0,Ke2];

%----- CASE EN COMPROBACION DEPTAGAS -----
gm1=1/15;
gm2=gm1;
gama=[gm1,0;0,gm2];
lmbd1=1.5;
lmbd2=lmbd1;
Kdv1=1.5;
Kdv2=0.5;

```

**Figura A-1.** Subrutina ini.m que almacena todas las constantes como: masa, distancia o largo del brazo, fricción, gravedad, etc. utilizadas en las funciones.

```

----- Este valor es para el control adaptativo -----
kdv=[5,0;0,5];
epsilon=0.1;

percent= 0.95;

----- Este valor es para el control adaptativo de los parámetros -----
kq1= 10.5;
kq2= 10.2;
kq = [kq1,0;0,kq2];
    
```

Figura A-1. (Cont.) Subrutina ini.m

```

Subrutina sal=td(u)

ini;
tt=u;
fact1= 1;
fact2= 0.5;
qd1= sin(fact1*tt);
qd2= cos(fact2*tt);
qdp1= fact1*cos(fact1*tt);
qdp2= -fact2*sin(fact2*tt);
qdpp1=-(fact1^2)*sin(fact1*tt);
qdpp2=-(fact2^2)*cos(fact2*tt);

sal=[qd1 qd2 qdp1 qdp2 qdpp1 qdpp2];
    
```

Figura A-2. Subrutina para simular las trayectorias deseadas para el estudio del seguimiento generado por los controladores, en este caso las trayectorias son senos y cosenos. El archivo es td.m

```

----- Este archivo muestra el error paramétrico y las masas estimadas
----- de la simulación del control adaptativo.

plot(t,eparam);
grid;
title('Error paramétrico');
figure
plot(t,salm);
grid;
title('Masas estimadas');
    
```

(a)

Figura A-3. Subrutina para imprimir. (a) los errores paramétricos y las masas, del control adaptable específicamente.

```

%----- las salidas -----
plot(t,e);
grid;
title('Posición de los errores: e1=q1 - q1d y e2=q2 - q2d');
figure
plot(t,ep);
grid;
title('Velocidades de cambio de los errores: ep1=q1d - q1 y ep2=q2d - q2');
figure
plot(t,salgen);
grid;
title('Todas las salidas generadas por el sistema controlado');

```

(b)

**Figura A-3.** (Cont.) Subrutina para imprimir: (b) las posiciones y velocidades de los errores, así como las salidas generadas por todos los controladores simulados en Matlab.

```

%----- salida=controlado -----
ini;
tetal= u(1);
teta2= u(2);
dtetal= u(3);
dteta2= u(4);
e1= u(5);
e2= u(6);
ep1=u(7);
ep2=u(8);
ddtetad1=u(9);
ddtetad2=u(10);
dtetad1=u(11);
dtetad2=u(12);
tetad1=u(13);
tetad2=u(14);
dteta=[dtetal;dteta2];
%----- M2= m2 -----
m11=(m1+m2)*(a1^2)+(m2*(a2^2))+(2*m2*a1*a2*cos(teta2));
m12=(m2*(a2^2))+(m2*a1*a2*cos(teta2));
m21=m12;
m22=m2*(a2^2);
%----- M -----
M=[m11,m12;m21,m22];
%----- V1= -2*m2*a1*a2*dteta2*sin(teta2) -----
v11=-2*m2*a1*a2*dteta2*sin(teta2);
v12=-m2*a1*a2*dteta2*sin(teta2);
v21=m2*a1*a2*dtetal*sin(teta2);
v22=0;
%----- V -----
V=[v11,v12;v21,v22];
C=V*dteta;
%----- g1= (m1+m2)*g*a1*cos(tetal)+(m2*g*a2*cos(tetal+teta2)); -----
g1=(m1+m2)*g*a1*cos(tetal)+(m2*g*a2*cos(tetal+teta2));

```

**Figura A-4.** Esta función esta diseñada para simular el control PAR CALCULADO.

```

g2=m2*g*a2*cos(tetal+teta2);
G=[g1;g2];
f1=(vis1*dtetal)+(k1*tanh(dtetal));
f2=(vis2*dteta2)+(k2*tanh(dteta2));
F=[f1;f2];
N=C+G+F;
sum1= ddtetad1 + ky1*ep1 + kp1*el;
sum2= ddtetad2 + ky2*ep2 + kp2*el;
sum = [sum1;sum2];
par = (M*sum)+N;
salidai=[par(1,1) par(2,1) tetad1 tetad2];

```

Figura A-4. (Cont.) Control PAR CALCULADO.

```

funcion salidai=cont2(u,
ini;
tetal= u(1);
teta2= u(2);
dtetal= u(3);
dteta2= u(4);
e1= u(5);
e2= u(6);
ep1=u(7);
ep2=u(8);
ddtetad1=u(9);
ddtetad2=u(10);
dtetad1= u(11);
dtetad2= u(12);
tetad1=u(13);
tetad2=u(14);
dtetad=[dtetad1;dtetad2];
ddtetad=[ddtetad1;ddtetad2];
m11=((m1+m2)*(a1^2))+(m2*(a2^2))+(2*m2*a1*a2*cos(teta2));
m12=(m2*(a2^2))+(m2*a1*a2*cos(teta2));
m21=m12;
m22=m2*(a2^2);
M=[m11,m12;m21,m22];
v11=-2*m2*a1*a2*dteta2*sin(teta2);
v12=-m2*a1*a2*dteta2*sin(teta2);
v21=m2*a1*a2*dtetal*sin(teta2);
v22=0;

```

Figura A-5. Esta función esta diseñada para simular el control PD.

```

----- CONTROL PID -----
V=[v11,v12;v21,v22];
C=V*dtetad;

----- CALCULO DE FUERZAS EN LOS MIEMBROS DE LA PUNTA -----
g1=((m1+m2)*g*a1*cos(tetal))+m2*g*a2*cos(tetal+teta2));
g2=m2*g*a2*cos(tetal+teta2);

----- FUERZA DE GRAVEDAD -----
G=[g1;g2];

----- FUERZAS DE RESORTE Y AMORTIGUAMIENTO -----
f1=(vis1*dtetad1)+(k1*tanh(dtetad1));
f2=(vis2*dtetad2)+(k2*tanh(dtetad2));

----- FUERZA DE CONTROL -----
F=[f1;f2];

e= [e1;e2];
ep=[ep1;ep2];

sum= (M*ddtetad)+G+e;
par= sum+(Kp*e)+(Kv*ep);

salida2=[par(1,1) par(2,1) tetad1 tetad2];

----- CONTROL PID -----
function salida3=cont3(u)
ini;

tetal= u(1);
teta2= u(2);
dtetal= u(3);
dteta2= u(4);
e1= u(5);
e2= u(6);
ep1= u(7);
ep2= u(8);
ddtetad1= u(9);
ddtetad2= u(10);
dtetad1= u(11);
dtetad2= u(12);
tetad1= u(13);
tetad2= u(14);
E1= u(15);
E2= u(16);

----- CONTROL PID -----
dtetad=[dtetad1;dtetad2];
ddtetad=[ddtetad1;ddtetad2];

----- CALCULO DE FUERZAS EN LOS MIEMBROS DE LA PUNTA -----
m11=((m1+m2)*(a1^2))+m2*(a2^2)+(2*m2*a1*a2*cos(teta2));
m12=(m2*(a2^2))+m2*a1*a2*cos(teta2);
m21=m12;
m22=m2*(a2^2);

----- MATRIZ DE INERCIA -----
M=[m11,m12;m21,m22];

```

Figura A-5. (Cont.) Control PD.

Figura A-6. Esta función esta diseñada para simular el control PID.

```

v11=-2*m2*a1*a2*dteta2*sin(teta2);
v12=-m2*a1*a2*dteta2*sin(teta2);
v21=m2*a1*a2*dteta1*sin(teta2);
v22=0;

V=[v11,v12;v21,v22];
C=V*dtetad;

g1=((m1+m2)*g*a1*cos(teta1))+(m2*g*a2*cos(teta1+teta2));
g2=m2*g*a2*cos(teta1+teta2);

G=[g1;g2];

f1=(vis1*dteta1)+(k1*aah*dteta1);
f2=(vis2*dteta2)+(k2*aah*dteta2);

F=[f1;f2];

e=[e1;e2];
ep=[ep1;ep2];
E=[E1;E2];

sum=(M*ddtetad)-C-G+F;
par=sum+(kp*e)+(ki*ep)+(kd*E);

salida3=[par(1,1) ps(1,1) [teta1 teta2];

salida4=cont4;

ini;

teta1= u(1);
teta2= u(2);
dteta1= u(3);
dteta2= u(4);
e1= u(5);
e2= u(6);
ep1=u(7);
ep2=u(8);
ddtetad1=u(9);
ddtetad2=u(10);
dteta1= u(11);
dteta2= u(12);
teta1=u(13);
teta2=u(14);

dteta =[dteta1;dteta2];
dtetad =[dteta1;dteta2];
ddtetad=[ddtetad1;ddtetad2];

m11=((m1+m2)*(a1^2))+(m2*(a2^2))+(2*m2*a1*a2*cos(teta2));
m12=(m2*(a2^2))+(m2*a1*a2*cos(teta2));
m21=m12;

```

Figura A-6. (Cont.) Control PID.

Figura A-7. Esta función esta diseñada para simular el control ESTRUCTURA PASIVA.

```

m22=m2*(a2^2);
M=[m11,m12;m21,m22];

v11=-2*m2*a1*a2*dteta2*sin(teta2);
v12=-m2*a1*a2*dteta2*sin(teta2);
v21=m2*a1*a2*dteta1*sin(teta2);
v22=0;

V=[v11,v12;v21,v22];

g1=((m1+m2)*g*a1*cos(teta1)+(m2*g*a2*cos(teta1+teta2)));
g2=m2*g*a2*cos(teta1+teta2);

G=[g1;g2];

f1=(vis1*dteta1)+(K1*tanh(dteta1));
f2=(vis2*dteta2)+(K2*tanh(dteta2));

F=[f1;f2];

e=[e1;e2];
ep=[ep1;ep2];
ddtetar=ddtetad+(LAMBDA*ep);
dtetar= dtetad+(LAMBDA*e);
s= dtetar-dteta;

C=V*dtetar;
sum= (M*ddtetar)+G+C+F;
par= sum+(Ke*s);

salida=[par(1), par(2), teta1, teta2];

```

**Figura A-7. (Cont.)** Control ESTRUCTURA PASIVA.

```

salida5=cont5(u)
ini;

teta1= u(1);
teta2= u(2);
dteta1= u(3);
dteta2= u(4);
e1= u(5);
e2= u(6);
ep1=u(7);
ep2=u(8);
ddtetad1=u(9);
ddtetad2=u(10);
dtetad1= u(11);
dtetad2= u(12);
tetad1= u(13);
tetad2= u(14);
mest1= u(15);
mest2= u(16);

```

**Figura A-8.** Esta función esta diseñada para simular el control ADAPTABLE.

```

Ya11=(a1^2)*(ddtetad1+(lmbd1*ep1))+g*a1*cos(tetal);

Ya12=((a1^2)+(a2^2)+2*a1*a2*cos(teta2))*(ddtetad1+(lmbd1*ep1))+((a2^2)+a1*a2*cos(teta2))
*(ddtetad2+(lmbd2*ep2))-(2*a1*a2*dteta2*sin(teta2))*(dtetad1+(lmbd1*ep1))-
(a1*a2*dteta2*sin(teta2))*(dtetad2+(lmbd2*ep2))+g*a1*cos(tetal)+g*a2*cos(tetal+teta2);
Ya21=0;

Ya22=((a2^2)+a1*a2*cos(teta2))*(ddtetad1+(lmbd1*ep1))+((a2^2)*(ddtetad2+(lmbd2*ep2)))+(a1*
a2*dtetal*sin(teta2))*(dtetad1+(lmbd1*ep1))+g*a2*cos(tetal+teta2);

Ya=[Ya11,Ya12;Ya21,Ya22];

mest=[mest1;mest2];

f1=(vis1*dtetal)+(K1*tan(dtetal));
f2=(vis2*dteta2)+(K2*tan(dteta2));

F=[f1;f2];

dtetad=[dtetad1;dtetad2];
dteta=[dtetal;dteta2];
tetad=[tetad1;tetad2];

e=[e1;e2];
ep=[ep1;ep2];
s=[(lmbd1*ep1+(lmbd2*ep2)+stet)];
sum= [(Kdv1,0;0,Kdv2)*s];

par= Ya*mest;sum;

salida5=[par(1,1) par(2,1) tetad(1,1) tetad(2,1)];

salm=estima(u)

ini;

tetal= u(1);
teta2= u(2);
dtetal= u(3);
dteta2= u(4);
e1= u(5);
e2= u(6);
ep1=u(7);
ep2=u(8);
ddtetad1=u(9);
ddtetad2=u(10);
dtetad1= u(11);
dtetad2= u(12);
tetad1= u(13);
tetad2= u(14);
mest1= u(15);
mest2= u(16);

```

Figura A-8. (Cont.) Control ADAPTABLE.

Figura A-9. Esta función esta diseñada para simular el ALGORITMO DE ESTIMACIÓN DE PARÁMETROS del Control Adaptable.

```

-----
Ya11=(a1^2)*(ddtetad1+(lmbd1*ep1))+g*a1*cos(tetal);
Ya12=((a1^2)+(a2^2)+2*a1*a2*cos(teta2))*(ddtetad1+(lmbd1*ep1))+((a2^2)+a1*a2*cos(teta2))
*(ddtetad2+(lmbd2*ep2))-(2*a1*a2*dteta2*sin(teta2))*(dtetad1+(lmbd1*el))-
(a1*a2*dteta2*sin(teta2))*(dtetad2+(lmbd2*ep2))+g*a1*cos(tetal)+g*a2*cos(tetal+teta2);
Ya21=0;
Ya22=((a2^2)+a1*a2*cos(teta2))*(ddtetad1+(lmbd1*ep1))+((a2^2)*(ddtetad2+(lmbd2*ep2)))+(a1*
a2*dtetal*sin(teta2))*(dtetad1+(lmbd1*el))+g*a2*cos(tetal+teta2);
-----
Ya=[Ya11,Ya12;Ya21,Ya22];
-----

s= [(lmbd1*el+ep1);(lmbd2*ep2+ep2)];

mestp=gama*(ya,'s');

salh=[mestp(1,1) mestp(2,1)];
-----

```

**Figura A-9.** (Cont.) ALGORITMO DE ESTIMACIÓN DE PARÁMETROS del Control Adaptable.

```

función salida6=cont6(u)
ini;

tetal= u(1);
teta2= u(2);
dtetal= u(3);
dteta2= u(4);
el= u(5);
e2= u(6);
ep1=u(7);
ep2=u(8);
ddtetad1=u(9);
ddtetad2=u(10);
dtetad1= u(11);
dtetad2= u(12);
tetad1= u(13);
tetad2= u(14);
-----
Ya11=(a1^2)*(ddtetad1+(lmbd1*ep1))+g*a1*cos(tetal);

Ya12=((a1^2)+(a2^2)+2*a1*a2*cos(teta2))*(ddtetad1+(lmbd1*ep1))+((a2^2)+a1*a2*cos(teta2))
*(ddtetad2+(lmbd2*ep2))-(2*a1*a2*dteta2*sin(teta2))*(dtetad1+(lmbd1*el))-
(a1*a2*dteta2*sin(teta2))*(dtetad2+(lmbd2*ep2))+g*a1*cos(tetal)+g*a2*cos(tetal+teta2);
Ya21=0;
Ya22=((a2^2)+a1*a2*cos(teta2))*(ddtetad1+(lmbd1*ep1))+((a2^2)*(ddtetad2+(lmbd2*ep2)))+(a1*
a2*dtetal*sin(teta2))*(dtetad1+(lmbd1*el))+g*a2*cos(tetal+teta2);
-----
Ya=[Ya11,Ya12;Ya21,Ya22];
-----

```

**Figura A-10.** Esta función esta diseñada para simular el control ROBUSTO con compensación a la incertidumbre paramétrica.

```

f1=(vis1*dteta1)+(k1*tanh(dteta1));
f2=(vis2*dteta2)+(k2*tanh(dteta2));
-----
F=[f1;f2];
-----
dtetad=[dtetad1;dtetad2];
dteta=[dteta1;dteta2];
tetad=[tetad1;tetad2];
-----
e = [e1;e2];
ep= [ep1;ep2];
s= [(lmbd1*e1+ep1);(lmbd2*e2+ep2)];
fi = [m1;m2];
fio= percent*fi;
efi= fi-fio;
p = norm(efi);
pro= (Ya.1)*s;
-----
if norm(pfo) > epsilon
mu=p*(pro/norm(pro));
norm(pfo) <= epsilon
mu=p*(pro/epsilon);
-----
sum= (kdv*s)+F;
-----
par= var((fi+bu)/sum;
-----
salida6=[par(1,1) par(2,1) tetad(1,1) tetad(2,1)];
-----
-----
salida7=cont7(u
ini;
-----
tetal= u(1);
teta2= u(2);
dtetal= u(3);
dteta2= u(4);
e1= u(5);
e2= u(6);
ep1=u(7);
ep2=u(8);
ddtetad1=u(9);
ddtetad2=u(10);
dtetad1= u(11);
dtetad2= u(12);
tetad1= u(13);
tetad2= u(14);
-----
dtetad=[dtetad1;dtetad2];
ddtetad=[ddtetad1;ddtetad2];
dteta=[dtetal;dteta2];
-----
m11=((m1+m2)*(a1^2))+(m2*(a2^2))+(2*m2*a1*a2*cos(teta2));

```

Figura A-10. (Cont.) Control ROBUSTO con compensación a la incertidumbre paramétrica.

Figura A-11. Esta función esta diseñada para simular el control de modo DESLIZANTE primera opción.

```

m12=(m2*(a2^2))+(m2*a1*a2*cos(teta2));
m21=m12;
m22=m2*(a2^2);

M=[m11,m12;m21,m22];

v11=-2*m2*a1*a2*dteta2*sin(teta2);
v12=-m2*a1*a2*dteta2*sin(teta2);
v21=m2*a1*a2*dteta1*sin(teta2);
v22=0;

V=[v11,v12;v21,v22];
C=V*dteta;

g1=((m1+m2)*g*a1*cos(teta1))+(m2*g*a2*cos(teta1+teta2));
g2=m2*g*a2*cos(teta1+teta2);

G=[g1;g2];

f1=(vis1*teta1)+(k1*tanh(dteta1));
f2=(vis2*dteta2)+(k2*tanh(dteta2));

F=[f1;f2];

N=C+G+F;

e= [e1;e2];
ep=[ep1;ep2];
LAMBDA=[lmbd1,0;0,lmbd2];
ddtetar=ddtetad+(LAMBDA*ep);
dtetar= dtetad+(LAMBDA*ep);
s= dtetar-dteta;

alfa= ddtetad+(LAMBDA*ep);
sum= (kq*tanh(s))+alfa;
par= (M*sum)+N;

salida7=[par(1,1) par(2,1) tetad1 tetad2];

```

Figura A-11. (Cont.) Control de modo DESLIZANTE primera opción.

```

salida8=cont71(u);
ini;

teta1= u(1);
teta2= u(2);
dteta1= u(3);
dteta2= u(4);
e1= u(5);
e2= u(6);
ep1=u(7);
ep2=u(8);
ddtetad1=u(9);
ddtetad2=u(10);
dtetad1= u(11);
dtetad2= u(12);
tetad1= u(13);
tetad2= u(14);

```

Figura A-12. Esta función esta diseñada para simular el control de modo DESLIZANTE segunda opción.

```

-----
m11=((m1+m2)*(a1^2))+(m2*(a2^2))+(2*m2*a1*a2*cos(teta2));
m12=(m2*(a2^2))+(m2*a1*a2*cos(teta2));
m21=m12;
m22=m2*(a2^2);

-----
M=[m11,m12;m21,m22];
-----
v11=-2*m2*a1*a2*dteta2*sin(teta2);
v12=-m2*a1*a2*dteta2*sin(teta2);
v21=m2*a1*a2*dteta1*sin(teta2);
v22=0;

-----
V=[v11,v12;v21,v22];

-----
g1=((m1+m2)*g*a1*cos(teta1))+m1*(g1a4*cos(teta1-teta2));
g2=m2*g*a2*cos(teta1+teta2);

-----
G=[g1;g2];

-----
f1=(vis1*dteta1)+k1*sign(dteta1);
f2=(vis2*dteta2)+k2*sign(dteta2);

-----
F=[f1;f2];

-----
dtetad=[dteta1;dteta2];
ddtetad=[ddtetad1;ddtetad2];
dteta=[dteta1;dteta2];

-----
e=[e1;e2];
ep=[ep1;ep2];
LAMBDA=[lmbd1,0;0,lmbd2];
ddtetar=ddtetad+(LAMBDA*ep);
dtetar=dtetad+(LAMBDA*e);
s=dtetar-dteta;

-----
C=V*(dtetar);

-----
beta=C+G+F;

-----
-----
alfa=ddtetar;
sum=(kq*tanh(s))+M*alfa;
par=(sum)+beta;

-----
-----
salida8=[par(1,1) par(2,1) tetad1 tetad2];

```

**Figura A-12. (Cont.) Control de modo DESLIZANTE segunda opción.**

APÉNDICE B

*Matlab*

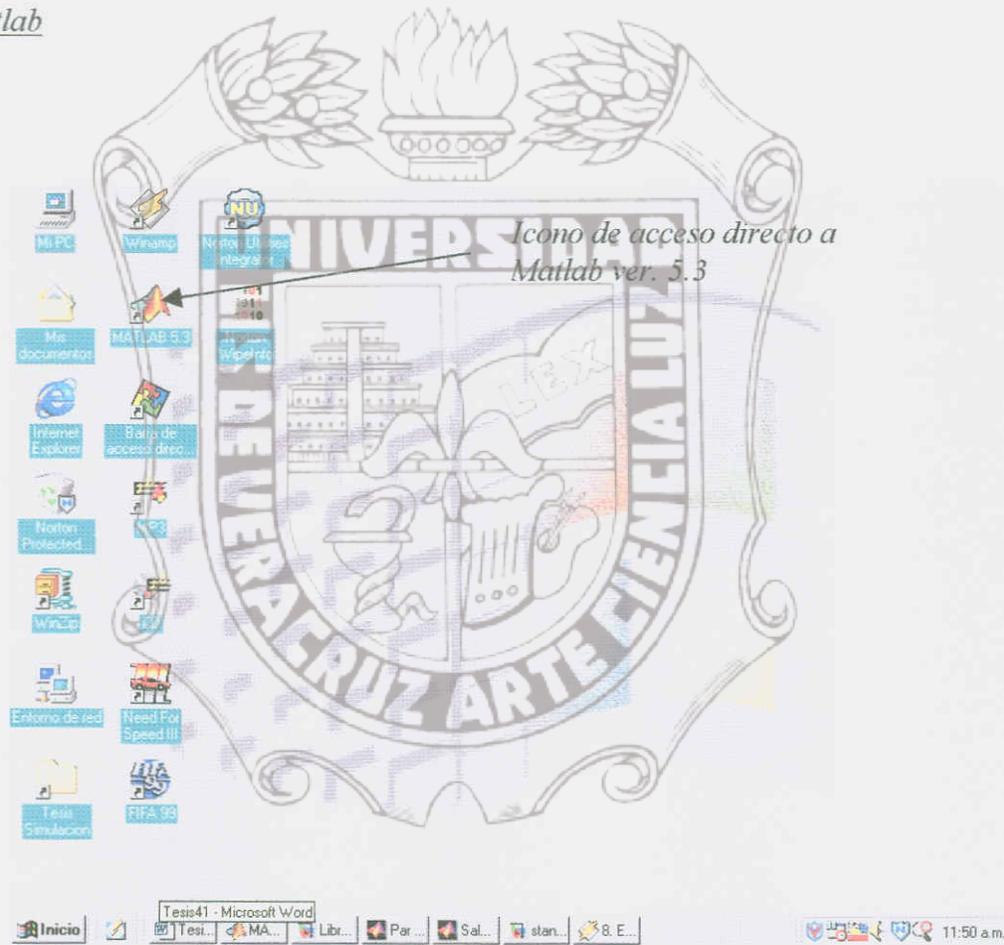


Figura B-1. Inicio rápido de Matlab en el escritorio de Windows.

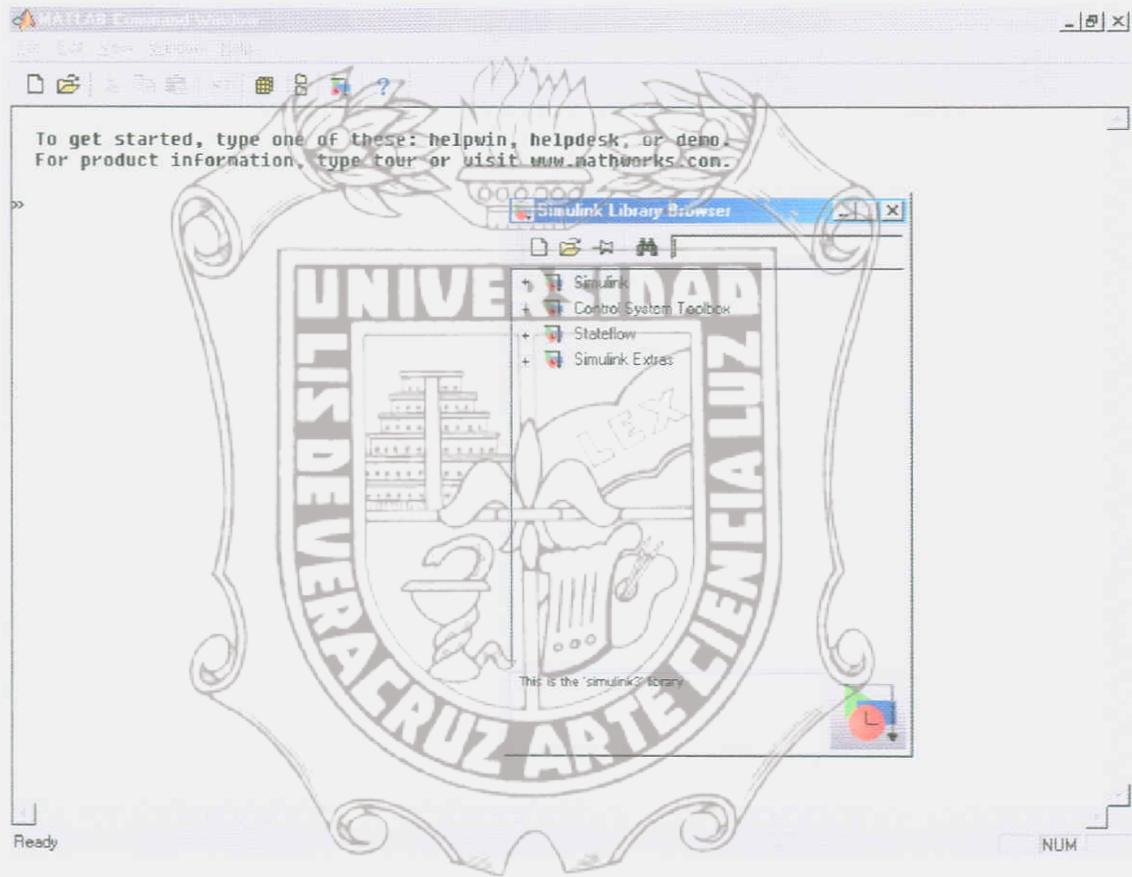
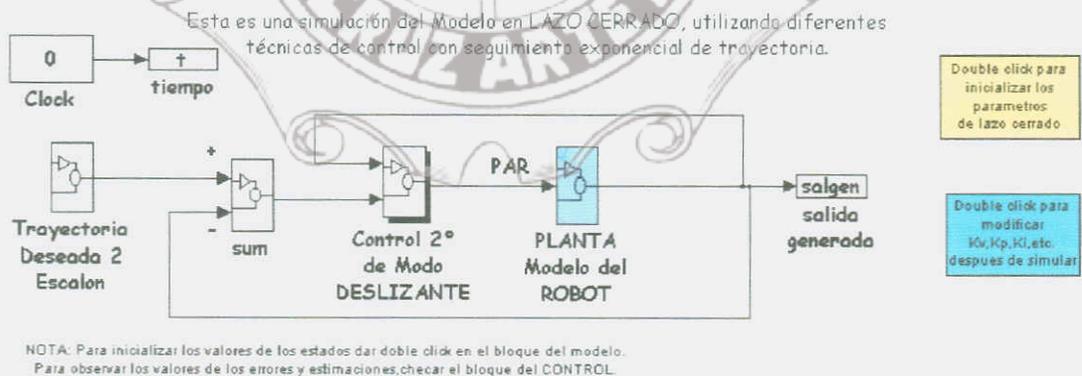


Figura B-2. Ventana de inicio en Matlab y SIMULINK ver. 3.0.

Diagramas utilizados en SIMULINK (Matlab).



Figura B-3 Sistema en Lazo Abierto.



NOTA: Para inicializar los valores de los estados dar doble click en el bloque del modelo.  
Para observar los valores de los errores y estimaciones, checar el bloque del CONTROL

Figura B-4. Sistema en Lazo Cerrado.

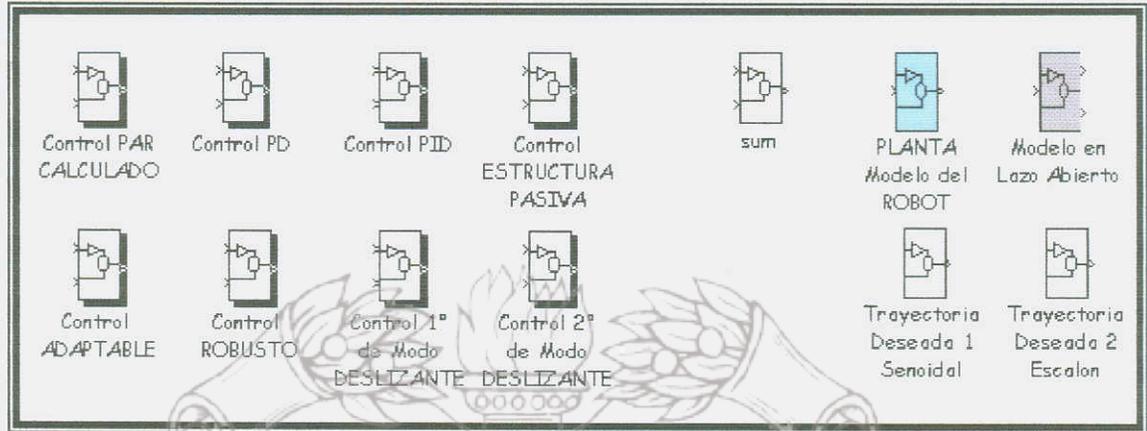
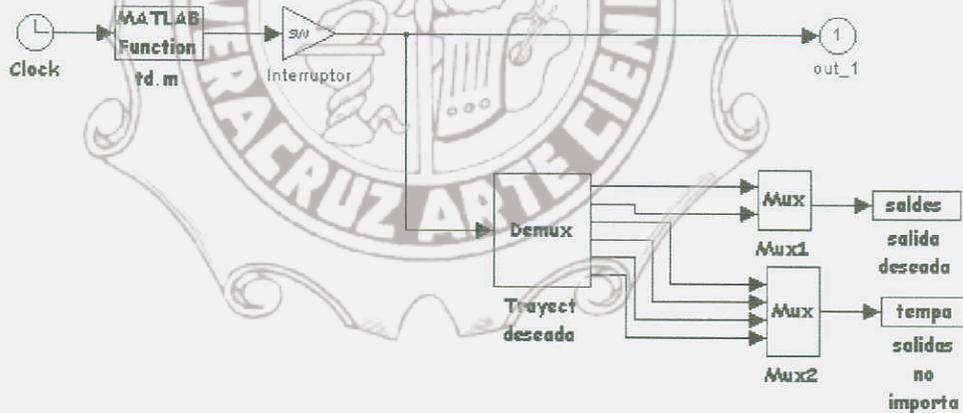


Figura B-5. Librería diseñada para operar diferentes controladores.

A continuación se muestra el contenido de los diferentes bloques diseñados, como lo son los sumadores, los de control, los de trayectoria deseada, etc:



NOTA: Este demux se utiliza para fines de separación de trayectoria y graficarlos (saldes) con respecto a las salidas generadas.

Las 4 trayectorias restantes se etiquetan como: "Salidas no importa" dentro de una variable llamada temp.

Figura B-6. Contenido del bloque Trayectoria deseada 1 Senoidal. Estas funciones senoidales son definidas en la figura A-2.

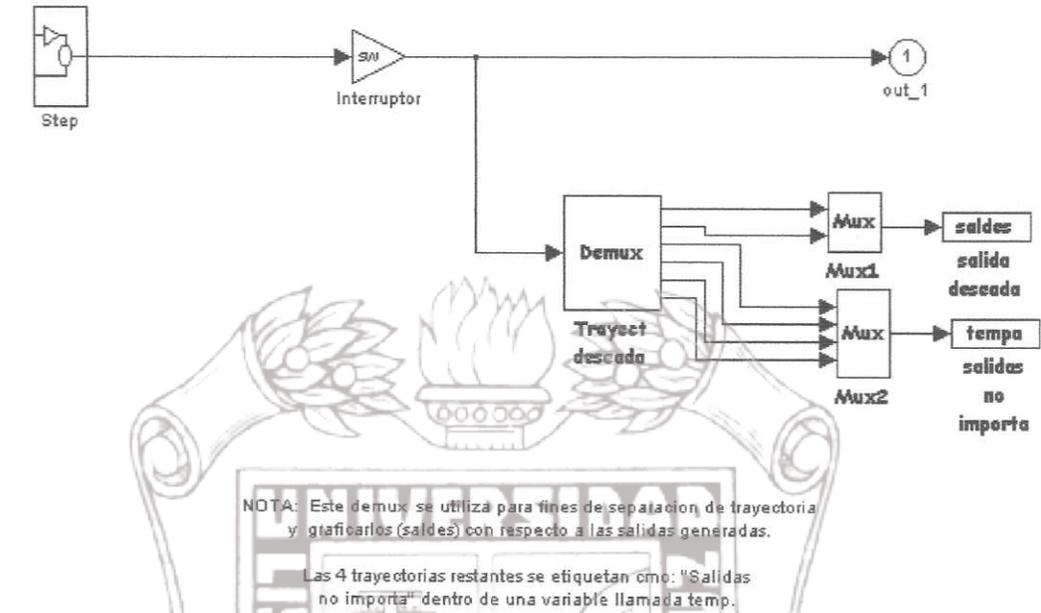


Figura B-7. Contenido del bloque Trayectoria deseada 2 Escalón.

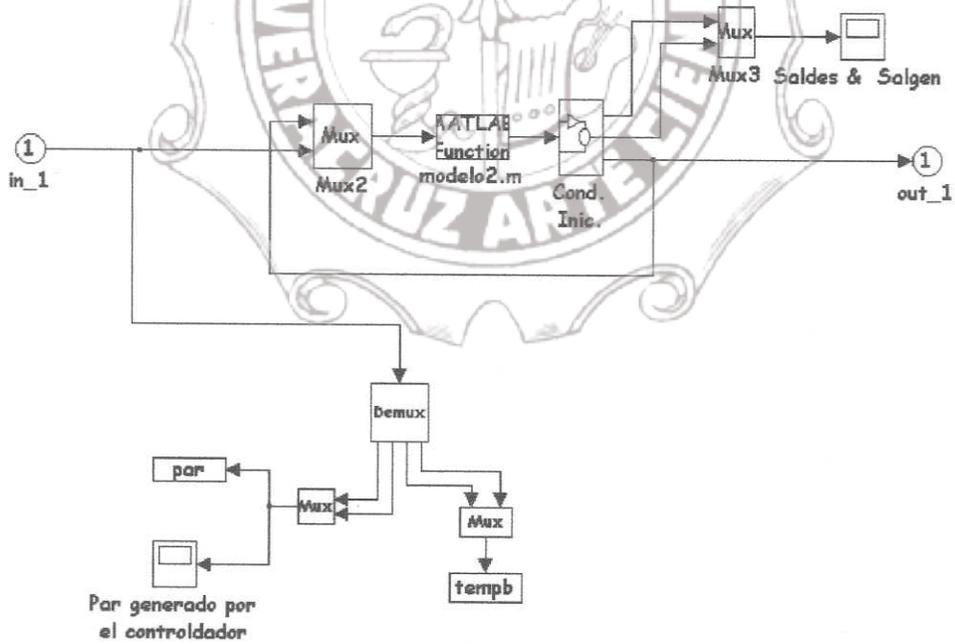


Figura B-8. Contenido del bloque PLANTA Modelo del ROBOT.

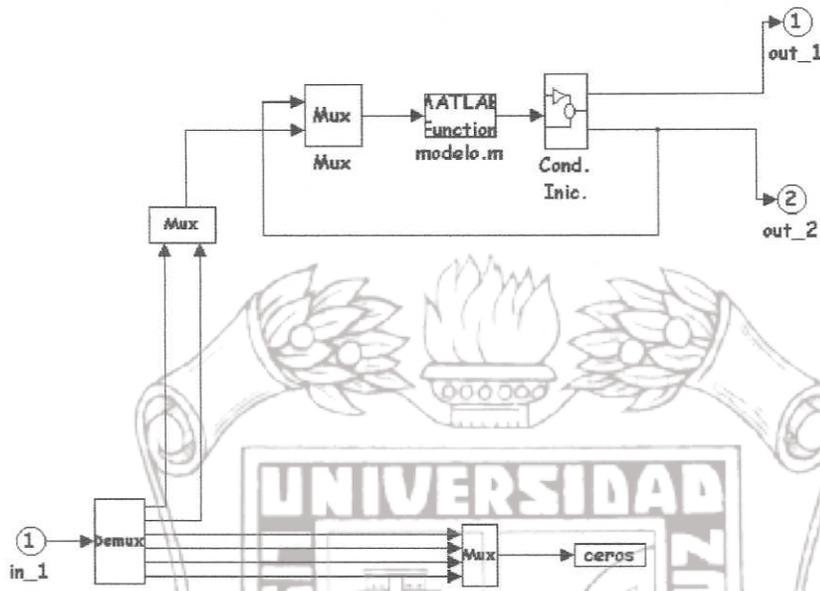


Figura B-9. Contenido del bloque Modelo Lazo Abierto.

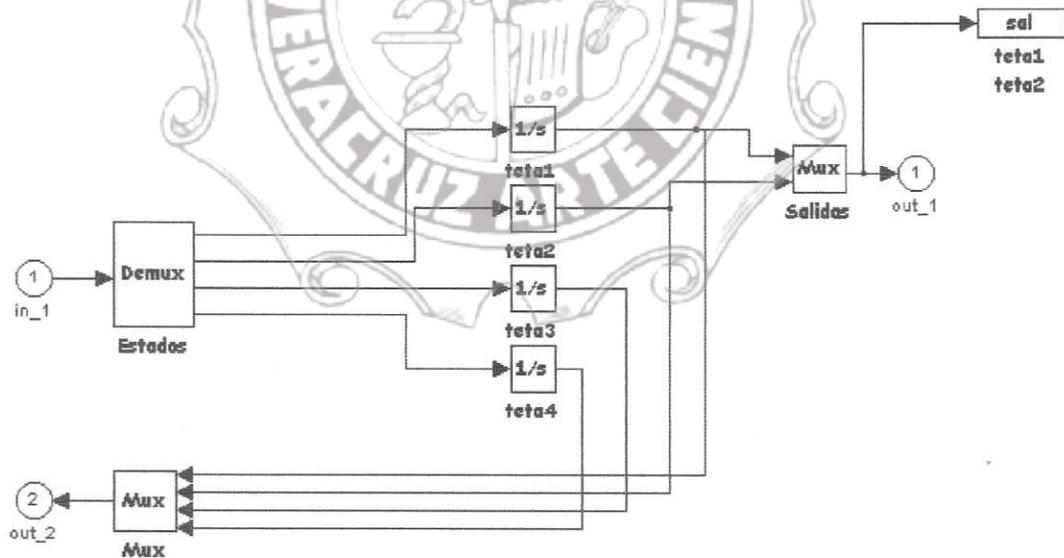


Figura B-10. Contenido del bloque de Condiciones Iniciales (Cond. Inic.).

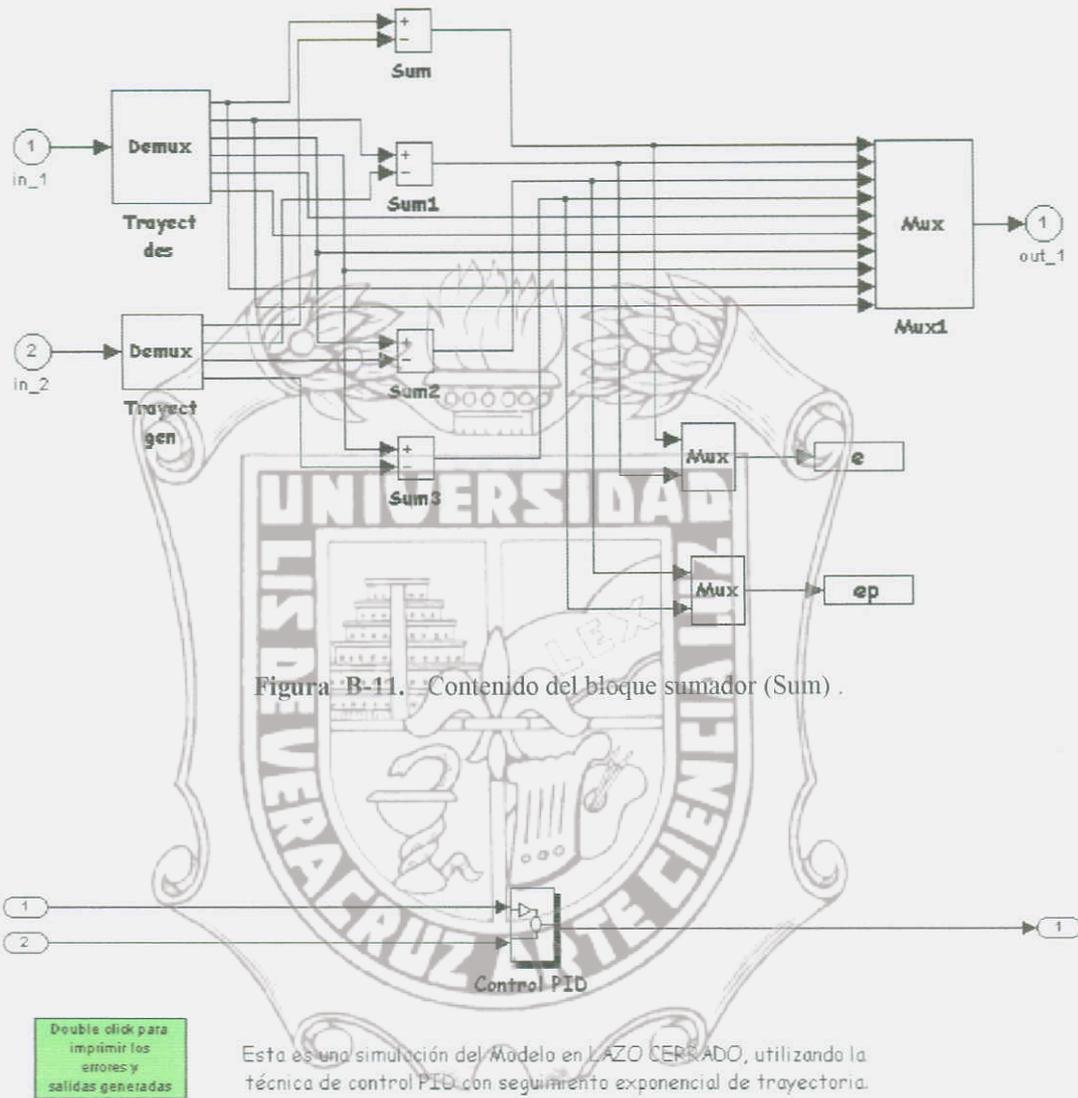


Figura B-12. Estructura interna en todos los bloques de control, para graficar los errores; excepto el Adaptable.

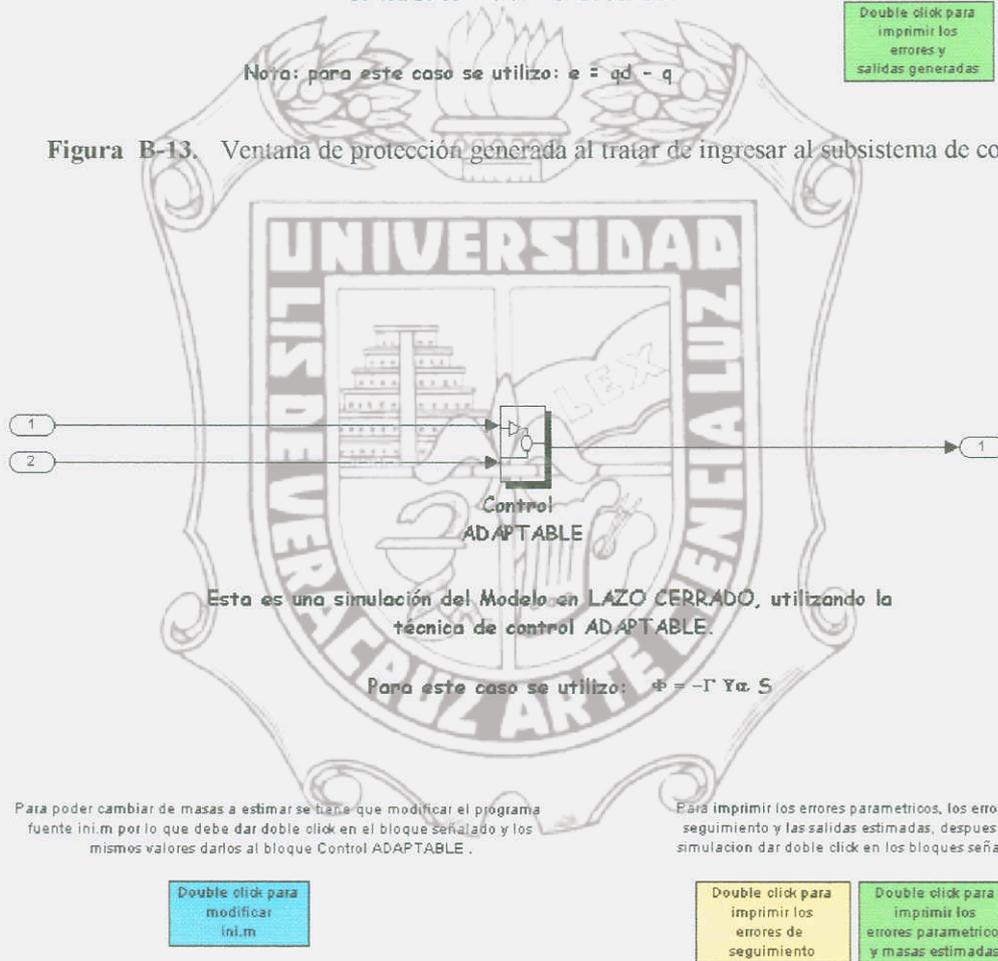


Esta es una simulación del Modelo en LAZ  
técnica de control PAR CALCULADO.

Nota: para este caso se utilizo:  $e = qd - q$

Double click para  
imprimir los  
errores y  
salidas generadas

Figura B-13. Ventana de protección generada al tratar de ingresar al subsistema de control.



Esta es una simulación del Modelo en LAZO CERRADO, utilizando la  
técnica de control ADAPTABLE.

Para este caso se utilizo:  $G = -1/Y \alpha S$

Para poder cambiar de masas a estimar se tiene que modificar el programa  
fuente ini.m por lo que debe dar doble click en el bloque señalado y los  
mismos valores darlos al bloque Control ADAPTABLE .

Para imprimir los errores parametricos, los errores de  
seguimiento y las salidas estimadas, despues de la  
simulación dar doble click en los bloques señalados.

Double click para  
modificar  
ini.m

Double click para  
imprimir los  
errores de  
seguimiento

Double click para  
imprimir los  
errores parametricos  
y masas estimadas

NOTA: PARA SIMULAR SE RECOMIENDA PONER UN TIEMPO DE INTEGRACION  
CON UN VALOR DE AL MENOS 50 PARA ESTE CONTROL.

Figura B-14. Estructura interna en bloques de control Adaptable, para graficar los errores.

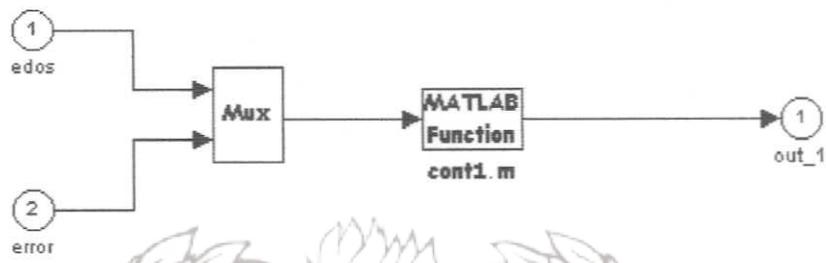


Figura B-15. Estructura interna para todos los subsistemas de control, excepto en control Adaptable.

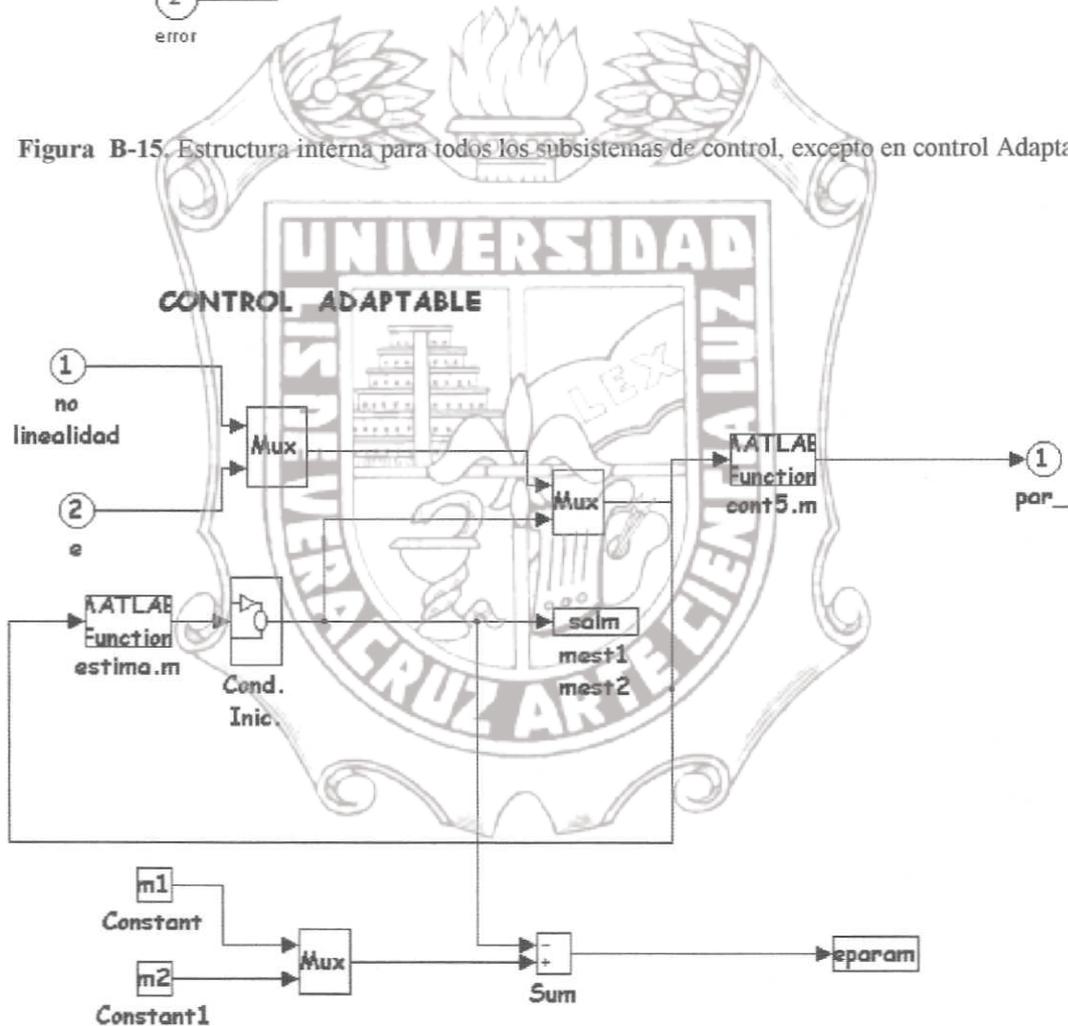


Figura B-16. Estructura interna para el subsistema del control Adaptable.

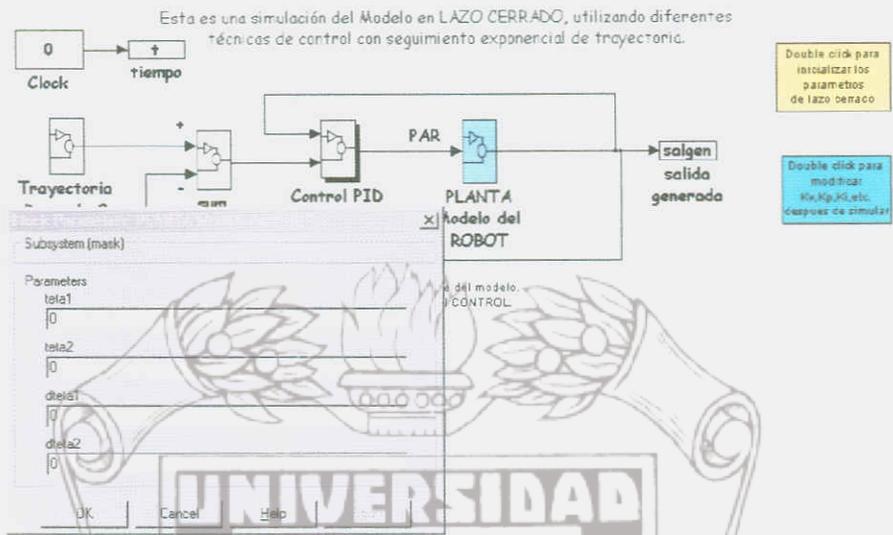


Figura B-17. Ventana generada para inicializar las variables de estado; esto se hace dando doble clic al bloque PLANTA Modelo del ROBOT (bloque color azul).

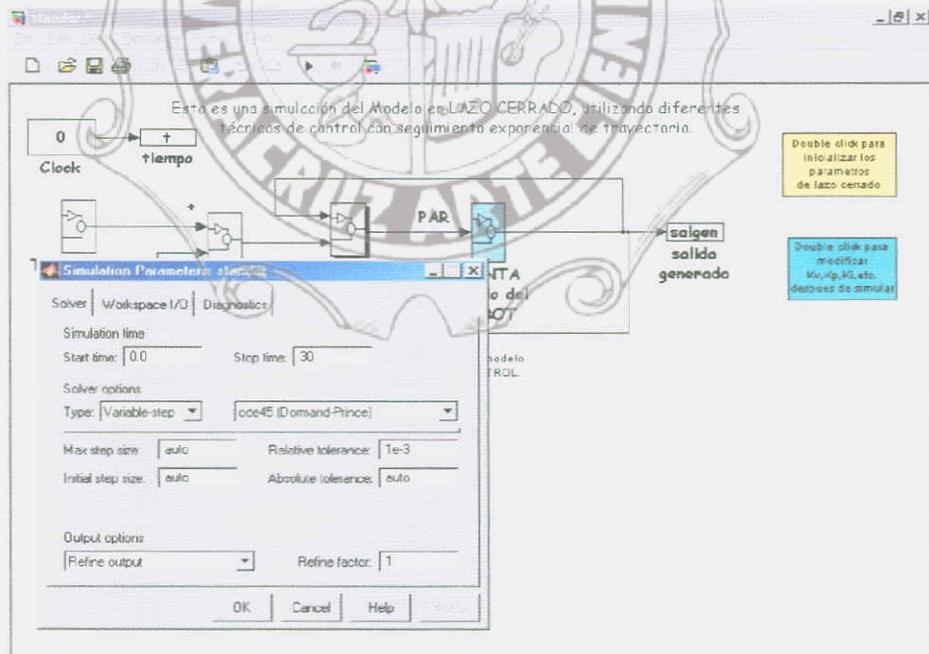


Figura B-18. Ventana de SIMULINK para dar parámetros de integración antes de la simulación.

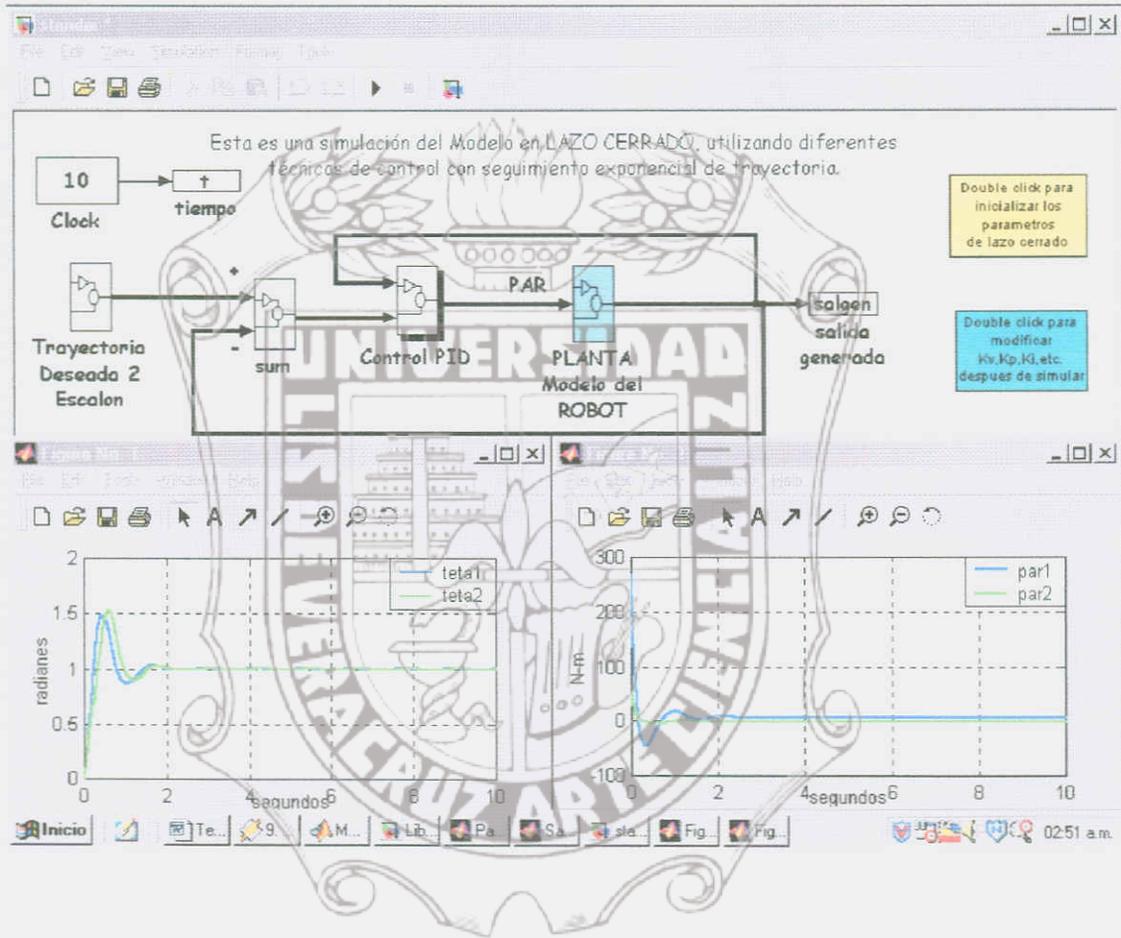


Figura B-19. Ambiente de trabajo en SIMULINK-Matlab en plena simulación.

## INDICE DE FIGURAS

### CAPITULO III

- Figura 3.1.** Varios tipos de articulaciones utilizadas en robots. 22
- Figura 3.2.** Diagramas a bloques del Espacio de Estado. 26

### CAPITULO IV

- Figura 4.1.** Brazo Planar de dos articulaciones. 35
- Figura 4.2.** Representación en variables de estado para sistemas no lineales. 40
- Figura 4.3.** Control de estructura pasiva. 62

### CAPITULO V

- Figura 5.1.** Simulación lazo abierto. 79
- Figura 5.2.** Simulación en lazo abierto con condiciones iniciales iguales a cero. 80
- Figura 5.3.** Simulación en lazo abierto con condiciones iniciales  $\theta_1 = \pi/2$  y  $\theta_2 = -\pi/2$ . 80
- Figura 5.4.** Simulación en lazo abierto con condiciones iniciales  $\theta_1 = 0$  y  $\theta_2 = \pi/2$ . 81
- Figura 5.5.** Simulación en lazo abierto con condiciones iniciales  $\theta_1 = -\pi/2$  y  $\theta_2 = 0$ . 81
- Figura 5.6.** Simulación en lazo abierto con condiciones iniciales  $\theta_1 = 0$  y  $\theta_2 = -\pi/2$ . 81

<b>Figura 5.7.</b> Simulación en lazo abierto con condiciones iniciales $\theta_1 = \pi/2$ y $\theta_2 = 0$ .	81
<b>Figura 5.8.</b> Subrutina para simular el modelo del manipulador usando Matlab.	82
<b>Figura 5.9.</b> Diagrama esquemático para controlar el robot manipulador.	83
<b>Figura 5.10.</b> Ángulos de las articulaciones $\theta_1$ y $\theta_2$ (rad) para el control PD-par-calculado, con un $kp = 100I$ , $kv = 20I$ .	84
<b>Figura 5.11.</b> Ángulos de las articulaciones $\theta_1$ y $\theta_2$ para el control PD, con un $kp = 100I$ , $kv = 20I$ .	84
<b>Figura 5.12.</b> Se puede observar las posiciones de los errores que tienden a cero, para el control PD-par-calculado, con un $kp = 100I$ , $kv = 20I$ .	84
<b>Figura 5.13.</b> Se puede observar las posiciones de los errores que tienden a cero, para el control PD, con un $kp = 100I$ , $kv = 20I$ .	84
<b>Figura 5.14.</b> Se puede observar las velocidades de cambio de los errores que tienden a cero, para el control PD-par-calculado, con un $kp = 100I$ , $kv = 20I$ .	85
<b>Figura 5.15.</b> Se puede observar las velocidades de cambio de los errores que tienden a cero, para el control PD, con un $kp = 100I$ , $kv = 20I$ .	85
<b>Figura 5.16.</b> Se puede observar el torque (N-m) generado por el control PD-par-calculado.	85
<b>Figura 5.17.</b> Se puede observar el torque (N-m) generado por el control PD.	85
<b>Figura 5.18.</b> Ángulos de las articulaciones $\theta_1$ y $\theta_2$ (rad). Se puede observar el seguimiento de las trayectorias generadas con las trayectorias deseadas (escalón unitario) para el control PID, con un $kp = 100I$ , $kv = 20I$ y $ki = 500I$ .	86

<b>Figura 5.19.</b> Ángulos de las articulaciones $\theta_1$ y $\theta_2$ (rad) para el control de ESTRUCTURA PASIVA.	86
<b>Figura 5.20.</b> Se puede observar las posiciones de los errores que tienden a cero, para el control PID, con un $k_p = 100I$ , $k_v = 20I$ y $k_i = 500I$ .	86
<b>Figura 5.21.</b> Se puede observar las posiciones de los errores que tienden a cero, para el control de ESTRUCTURA PASIVA, con un $k_e = 20I$ , $\lambda = 10I$ .	86
<b>Figura 5.22.</b> Se puede observar las velocidades de cambio de los errores que tienden a cero, para el control PID, con un $k_p = 100I$ , $k_v = 20I$ y $k_i = 500I$ .	87
<b>Figura 5.23.</b> Se puede observar las velocidades de cambio de los errores que tienden a cero, para el control ESTRUCTURA PASIVA, con un $k_e = 20I$ , $\lambda = 10I$ .	87
<b>Figura 5.24.</b> Se puede observar el torque (N-m) generado por el control PID.	87
<b>Figura 5.25.</b> Se puede observar el torque (N-m) generado por el control de ESTRUCTURA PASIVA.	87
<b>Figura 5.26.</b> Ángulos de las articulaciones $\theta_1$ y $\theta_2$ (rad). Se puede observar el seguimiento de las trayectorias generadas, para el control ADAPTABLE, con un $k = 1.5I$ , $\lambda = 1.5I$ y $\gamma = (1/15)I$ .	88
<b>Figura 5.27.</b> Se puede observar las posiciones de los errores que tienden a cero, para el control ADAPTABLE.	88
<b>Figura 5.28.</b> Se puede observar las velocidades de cambio de los errores que tienden a cero, para el control ADAPTABLE.	88
<b>Figura 5.29.</b> Estimación de parámetros $m_1=1$ y $m_2=0.5$ del control ADAPTABLE.	88
<b>Figura 5.30.</b> Errores paramétricos $\tilde{\varphi}_1$ y $\tilde{\varphi}_2$ del control ADAPTABLE.	89

<b>Figura 5.31.</b>	Torque generado por el control ADAPTABLE.	89
<b>Figura 5.32.</b>	Ángulos de las articulaciones $\theta_1$ y $\theta_2$ (rad) del control ROBUSTO.	89
<b>Figura 5.33.</b>	Ángulos de las articulaciones $\theta_1$ y $\theta_2$ (rad) para el control de modo DESLIZANTE.	89
<b>Figura 5.34.</b>	Se puede observar las posiciones de los errores que tienden a cero, para el control ROBUSTO.	90
<b>Figura 5.35.</b>	Se puede observar las posiciones de los errores que tienden a cero, para el control de modo DESLIZANTE.	90
<b>Figura 5.36.</b>	Se puede observar las velocidades de cambio de los errores que tienden a cero, para el control ROBUSTO.	90
<b>Figura 5.37.</b>	Se puede observar las velocidades de cambio de los errores que tienden a cero, para el control de modo DESLIZANTE.	90
<b>Figura 5.38.</b>	Torque generado por el control ROBUSTO.	91
<b>Figura 5.39.</b>	Torque generado por el control de modo DESLIZANTE.	91
<b>Figura 5.40.</b>	Ángulos de las articulaciones $\theta_1$ y $\theta_2$ (rad) para el control de modo DESLIZANTE (2ª opción)	91
<b>Figura 5.41.</b>	Se puede observar las posiciones de los errores que tienden a cero, para el control de modo DESLIZANTE (2ª opción)	91
<b>Figura 5.42.</b>	Torque generado por el control de modo DESLIZANTE (2ª opción)	92
<b>Figura 5.43.</b>	Ángulos de las articulaciones $\theta_1$ y $\theta_2$ (rad) para el control PD-par- calculado, con un $k_p = 100I$ , $k_v = 20I$ .	94
<b>Figura 5.44.</b>	Ángulos de las articulaciones $\theta_1$ y $\theta_2$ (rad) para el control PD, con un $k_p = 100I$ , $k_v = 20I$ . Este control varía porque está diseñado por Lyapunov.	93

<b>Figura 5.45.</b>	Se puede observar las posiciones de los errores que tienden a cero, para el control PD-par-calculado, con un $kp = 100I$ , $k_v = 20I$ .	93
<b>Figura 5.46.</b>	Se puede observar las posiciones de los errores que tienden a cero, para el control PD, con un $kp = 100I$ , $k_v = 20I$ .	93
<b>Figura 5.47.</b>	Se puede observar el torque (N-m) generado por el control PD-par-calculado.	94
<b>Figura 5.48.</b>	Se puede observar el torque (N-m) generado por el control PD.	94
<b>Figura 5.49.</b>	Ángulos de las articulaciones $\theta_1$ y $\theta_2$ (rad). Se puede observar el seguimiento de las trayectorias generadas con las trayectorias deseadas (escalón unitario) para el control PID, con un $kp = 100I$ , $k_v = 20I$ y $ki = 500I$ .	94
<b>Figura 5.50.</b>	Ángulos de las articulaciones $\theta_1$ y $\theta_2$ (rad) para el control de ESTRUCTURA PASIVA.	94
<b>Figura 5.51.</b>	Se puede observar las posiciones de los errores que tienden a cero, para el control PID, con un $kp = 100I$ , $k_v = 20I$ y $ki = 500I$ .	95
<b>Figura 5.52.</b>	Se puede observar las posiciones de los errores que tienden a cero, para el control de ESTRUCTURA PASIVA, con un $ke = 20I$ , $\lambda = 10I$ .	95
<b>Figura 5.53.</b>	Se puede observar el torque (N-m) generado por el control PID.	95
<b>Figura 5.54.</b>	Se puede observar el torque (N-m) generado por el control ESTRUCTURA PASIVA.	95
<b>Figura 5.55.</b>	Seguimiento de las trayectorias deseadas.	96
<b>Figura 5.56.</b>	Posición de los errores.	96
<b>Figura 5.57.</b>	Estimación de parámetros (masas $m_1$ y $m_2$ ).	96
<b>Figura 5.58.</b>	Par generado o señal de control.	96

<b>Figura 5.59.</b>	Seguimiento de las trayectorias deseadas. Control Robusto.	97
<b>Figura 5.60.</b>	Seguimiento de las trayectorias deseadas. Control Deslizante.	97
<b>Figura 5.61.</b>	Errores de posición. Control robusto.	97
<b>Figura 5.62.</b>	Errores de posición. Control Deslizante.	97
<b>Figura 5.63.</b>	Par generado por el control Robusto.	97
<b>Figura 5.64.</b>	Par generado por el control Deslizante.	97
<b>APÉNDICE A</b>		
<b>Figura A-1.</b>	Subrutina ini.m que almacena todas las constantes como: masa, distancia o largo del brazo, fricción, gravedad, etc. utilizadas en las funciones.	100
<b>Figura A-2.</b>	Subrutina para simular las trayectorias deseadas para el estudio del seguimiento generado por los controladores, en este caso las trayectorias son senos y cosenos. El archivo es td.m.	101
<b>Figura A-3.</b>	Subrutinas para imprimir: (a) los errores paramétricos y las masas, del control adaptable específicamente y (b) las posiciones y velocidades de los errores, así como las salidas generadas por todos los controladores simulados en Matlab.	101
<b>Figura A-4.</b>	Esta función esta diseñada para simular el control PAR CALCULADO.	102
<b>Figura A-5.</b>	Esta función esta diseñada para simular el control PD.	103
<b>Figura A-6.</b>	Esta función esta diseñada para simular el control PID.	104

<b>Figura A-7.</b>	Esta función esta diseñada para simular el control ESTRUCTURA PASIVA.	105
<b>Figura A-8.</b>	Esta función esta diseñada para simular el control ADAPTABLE.	106
<b>Figura A-9.</b>	Esta función esta diseñada para simular el ALGORITMO DE ESTIMACIÓN DE PARÁMETROS del Control Adaptable.	107
<b>Figura A-10.</b>	Esta función esta diseñada para simular el control ROBUSTO con compensación a la incertidumbre paramétrica.	108
<b>Figura A-11.</b>	Esta función esta diseñada para simular el control de modo DESLIZANTE primera opción.	109
<b>Figura A-12.</b>	Esta función esta diseñada para simular el control de modo DESLIZANTE segunda opción.	110
<b>APÉNDICE B</b>		
<b>Figura B-1.</b>	Inicio rápido de Matlab en el escritorio de Windows	112
<b>Figura B-2.</b>	Ventana de inicio en Matlab y SIMULINK ver. 3.0.	113
<b>Figura B-3</b>	Sistema en Lazo Abierto.	114
<b>Figura B-4.</b>	Sistema en Lazo Cerrado.	114
<b>Figura B-5.</b>	Librería diseñada para operar diferentes controladores.	115
<b>Figura B-6.</b>	Contenido del bloque Trayectoria deseada 1 Senoidal. Estas funciones senoidales son definidas en la figura A-2.	115
<b>Figura B-7.</b>	Contenido del bloque Trayectoria deseada 2 Escalón.	116

<b>Figura B-8.</b>	Contenido del bloque PLANTA Modelo del ROBOT.	116
<b>Figura B-9.</b>	Contenido del bloque Modelo Lazo Abierto.	117
<b>Figura B-10.</b>	Contenido del bloque de Condiciones Iniciales (Cond. Inic.).	117
<b>Figura B-11.</b>	Contenido del bloque sumador (Sum).	118
<b>Figura B-12.</b>	Estructura interna en todos los bloques de control, para graficar los errores; excepto el Adaptable.	118
<b>Figura B-13.</b>	Ventana de protección generada al tratar de ingresar al subsistema de control.	119
<b>Figura B-14.</b>	Estructura interna en bloques de control Adaptable, para graficar los errores.	119
<b>Figura B-15.</b>	Estructura interna para todos los subsistemas de control, excepto en control Adaptable.	120
<b>Figura B-16.</b>	Estructura interna para el subsistema del control Adaptable.	120
<b>Figura B-17.</b>	Ventana generada para inicializar las variables de estado; esto se hace dando doble clic al bloque PLANTA Modelo del ROBOT (bloque color azul).	121
<b>Figura B-18.</b>	Ventana de SIMULINK para dar parámetros de integración antes de la simulación.	121
<b>Figura B-19.</b>	Ambiente de trabajo en SIMULINK-Matlab en plena simulación.	122

## INDICE DE TABLAS

Tabla 3.1	Clasificación de los robots según la AFRI.	12
Tabla 3.2	Clasificación de los robots industriales en generaciones.	13
Tabla 3.3	Clasificación de los robots según T.M. Knasel	15'

