

Cálculo de la Función de Transferencia de un Robot Móvil Empleando Filtros Adaptables

Tesis presentada por

Rubén Anaya García

en el área de Electrónica de Potencia, campo de conocimiento de
Ingeniería Eléctrica

para obtener el grado de

Maestro

en

Ingeniería

en el

Programa de Maestría y Doctorado en Ingeniería

de la

Universidad Nacional Autónoma de México

Ciudad Universitaria, México D.F.

Director de Tesis:

Dr. Jesús Savage Carmona

2009



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

Esta tesis fue aprobada por el siguiente jurado:

Dr. Yu Tang Xu
Presidente

Dr. José Cohen Sak
Secretario

Dr. Jesús Savage Carmona
Primer Vocal

MI. Luis Arturo Haro Ruíz
Primer Suplente

Dr. Carlos Rivera Rivera
Segundo Suplente

Dr. Jesús Savage Carmona

Tutor

Universidad Nacional Autónoma de México, Ciudad Universitaria, México D.F.

2009

Con amor y cariño para Verónica, Rubén Alexis y Verónica Aleydis...

A Verónica por su paciencia y comprensión a todo lo que emprendo, por que tu más que nadie sabe lo he me ha costado perseguir y obtener mis metas, que también hago tuyas; a mis hijos que por momentos he dedicado más tiempo a mis actividades que estar con ustedes, pero que ahora que son mayores y ya he logrado interesarlos en lo que me ha apasionado desde hace mucho tiempo, podremos tener una afinidad más que nos una como familia.

A mis padres Herlinda y Lidio por su cariño y respeto que me muestran y que sin lugar a dudas ustedes me han puesto en el camino y fuí libre de seguirlo y no me arrepiento.

A mis hermanos Francisco, Ismael, Oscar y Marisol.

A mis sobrinos, sobrinas y cuñadas, que aunque no crean estoy al pendiente de lo que hacen.

A todos ustedes y al resto de la familia dedico mis esfuerzos.

Agradecimientos

Mi sincero agradecimiento a la Universidad Nacional Autónoma de México, que me ha forjado como estudiante, profesional y académico.

Un agradecimiento especial al Doctor Jesús Savage Carmona por todo el apoyo, tiempo y conocimiento que me ha brindado no solo en el aspecto académico, también en el laboral y el personal .

A los distinguidos académicos que forman parte del jurado, por sus atinados comentarios hechos a este trabajo y que en su mayoría fueron mis profesores en los estudios de maestría.

A mis alumnos y compañeros de trabajo.

Al Consejo Nacional de Ciencia y Tecnología CONACYT por la beca que tuvo a bien otorgarme durante la realización de mis estudios.

A la UNAM, a la FI, a la DIE, al Departamento de Ingeniería en Computación y a la Unidad de Estudios de Posgrado...!!

. Gracias por ayudarme a lograrlo...!!

Rubén Anaya García
Ciudad Universitaria, México, D.F., 2009

Resumen

En el presente trabajo se describen las características generales del empleo de filtrado adaptable en la modalidad de identificación de sistemas, para encontrar la función de transferencia de un robot móvil con estructura mecánica tipo diferencial, a través de la metodología experimental aplicable por los algoritmos adaptables; además se realizará el análisis matemático del modelo del robot para obtener un aproximado de la función de transferencia.

Al excitar robot móvil con una señal de entrada, considerada la señal deseada, se obtiene una respuesta que con seguridad no es estrictamente la esperada, por lo que para obtener un desempeño favorable es necesario conocer la función de transferencia del robot.

Calcular analíticamente la función de transferencia es una tarea difícil, debido a la cantidad de parámetros de sus componentes y las no linealidades de éstos; por lo tanto se propone la utilización de técnicas de identificación de sistemas para estimar el modelo.

Al ser conocida la señal de entrada (velocidad deseada) y la respuesta del robot (velocidad real), se observa que esta respuesta es dependiente de sus características internas; por lo tanto conociendo esta información que es obtenida experimentalmente así como la estructura del modelo, se puede hacer la estimación de los parámetros de la función de transferencia utilizando técnicas de filtrado adaptable.

Se diseña y construye el robot móvil para tener un prototipo de prueba.

Índice general

Índice de figuras	IV
Índice de cuadros	VI
1. Introducción	1
1.1. Motivación	1
1.2. Justificación	2
1.3. Presentación	2
1.4. Objetivos generales	4
1.5. Objetivos específicos	4
1.6. Metodología	5
1.7. Estructura del trabajo tesis	6
2. Filtros adaptables	7
2.1. Filtro de Wiener	8
2.2. Algoritmos adaptables	16
2.2.1. Algoritmo de máximo descenso	16
2.2.2. Algoritmo LMS para filtros no recursivos FIR	17
2.2.3. Algoritmo LMS para filtros recursivos IIR	19
2.3. Aplicaciones de filtrado adaptable	23
2.3.1. Predicción	23
2.3.2. Identificación de sistemas	25
2.3.3. Ecuación	25
2.3.4. Cancelación de interferencias	25
3. Modelado del robot	28
3.1. Dinámica del robot	28
3.1.1. Movimiento de las llantas	28
3.1.2. Movimiento lineal del robot	30
3.1.3. Movimiento rotacional del robot	31
3.2. Cinemática del robot	36

4. Robot móvil	40
4.1. Descripción del robot	42
4.2. Motores	43
4.3. Estructura física	45
4.4. Microcontrolador	46
4.5. Etapa de potencia	50
4.6. Sensores	50
4.6.1. Sensores de contacto	51
4.6.2. Infrarrojos	52
4.6.3. Sensor ultrasónico (sonar)	52
4.7. Software	54
4.7.1. Modulación de ancho de pulso	54
4.7.2. Captura de pulsos (encoders)	56
4.7.3. Interrupción (TIMER2)	57
4.7.4. Velocidad de entrada	58
4.7.5. Algoritmo recursivo PID	59
5. Estructura del filtro	62
5.1. Análisis de la estructura del modelo	62
5.2. Estimación de los parámetros	62
5.2.1. Estimación usando filtro FIR	63
5.2.2. Estimación usando filtro IIR	64
6. Pruebas y análisis	66
6.1. Realización de experimentos	66
6.2. Resultados	70
6.2.1. Respuesta estimando con un filtro FIR	70
6.2.2. Respuesta estimando con un filtro IIR	73
6.3. Validación	74
6.3.1. Validación filtro FIR	76
6.3.2. Validación filtro IIR	78
7. Conclusiones	82

Índice de figuras

1.1. Controlador	2
1.2. Representación de la identificación de sistemas	3
2.1. Filtro adaptable	8
2.2. Filtro FIR	17
2.3. Algoritmo LMS para un filtro adaptable FIR	20
2.4. Filtro adaptable IIR	20
2.5. Algoritmo LMS filtros IIR	24
2.6. Predictor	24
2.7. Identificación de sistemas	25
2.8. Ecualizador	26
2.9. Cancelador de interferencias	26
3.1. Robot móvil diferencial	29
3.2. Identificación general del robot	36
3.3. Trayectoria del robot	38
4.1. Robot móvil	40
4.2. Robot móvil visto desde sus cuatro lados	41
4.3. Robot móvil distribución diferencial	42
4.4. Motor Maxon	43
4.5. Microcontrolador PIC16F877	47
4.6. Comunicación serie entre la PC y un microcontrolador	48
4.7. Sistema mínimo microcontrolador PIC	49
4.8. Tarjeta usada PIC16F877	49
4.9. Control del circuito L298	51
4.10. Electrónica microswitch	52
4.11. Diagrama a bloques sensor GP2Y0A21	53
4.12. Distribución onda sonora	54
4.13. Modulación de ancho de pulso	55
4.14. Captura de pulsos	57
4.15. Rutina de interrupción TIMER2	58
4.16. Recepción de datos	59
4.17. Controlador PID	60

4.18. Algoritmo PID	61
5.1. Identificación de sistemas	63
5.2. Algoritmo LMS Filtro FIR	64
5.3. Algoritmo LMS Filtro IIR	65
6.1. Prototipo de prueba	67
6.2. Velocidad de entrada deseada motor izquierdo	68
6.3. Velocidad de entrada deseada motor derecho	69
6.4. Comportamiento del filtro adaptable motor izquierdo (FIR)	71
6.5. Comportamiento del filtro adaptable motor derecho (FIR)	72
6.6. Comportamiento del filtro adaptable motor izquierdo (IIR)	73
6.7. Comportamiento del filtro adaptable motor derecho (IIR)	75
6.8. Respuesta de la función encontrada para el motor izquierdo (FIR)	76
6.9. Respuesta de la función encontrada para el motor derecho (FIR)	77
6.10. Respuesta de la función encontrada para el motor izquierdo (IIR)	78
6.11. Respuesta de la función encontrada para el motor derecho (IIR)	79
6.12. Velocidad angular llanta izquierda	80
6.13. Velocidad angular llanta derecha	81

Índice de cuadros

6.1. Coeficientes filtro FIR llanta izquierda	70
6.2. Coeficientes filtro FIR llanta derecha	73
6.3. Coeficientes filtro IIR llanta izquierda	74
6.4. Coeficientes filtro IIR llanta derecha	74

Capítulo 1

Introducción

En años recientes ha existido un gran auge en aplicaciones de la robótica en las distintas áreas de conocimiento; utilizando robots móviles (con llantas o extremidades) o los robots manipuladores fijos. En ambos casos se requiere que el robot responda adecuadamente a las señales de control indicadas; para lo cual se necesita conocer el modelo del robot.

1.1. Motivación

La construcción de un robot móvil involucra una gran variedad de elementos a considerar; como es la elección de los motores de corriente directa a los cuales le son agregados mecanismos de reducción con lo que consigue bajar la velocidad de rotación pero se aumenta el torque; las llantas y rodamientos que se adaptan para tener apoyos adecuados a la superficie de deslizamiento, las bases y chasis para soportar el peso de sus componentes, el hardware encargado de generar el movimiento del robot, los distintos tipos de sensores, el microcontrolador y las pilas que serán utilizadas.

Encontrar la función de transferencia de un sistema requiere del conocimiento de todos sus parámetros, resultando en algunos casos una tarea difícil pero una vez conocida es de gran utilidad. El reto principal es utilizar técnicas que permitan conocer el modelo. Construir y controlar un robot móvil implica la interacción de varias disciplinas de la ingeniería, por lo que al utilizar técnicas de filtrado adaptable en sus diferentes aplicaciones y sobre todo en la identificación podrá ser de gran ayuda en proyectos actuales y futuros.

1.2. Justificación

Al tratarse de un sistema no lineal que complica el análisis, el robot móvil es aproximado a un sistema lineal y considerado como tal, haciendo posible el empleo de las técnicas de filtrado adaptable para estimar su función de transferencia, que permita describir la dinámica del robot sin necesidad de conocer los parámetros que en su conjunto lo integran; por lo que una vez estimada se podrá reproducir y simular su comportamiento y en caso de ser requerido sea factible utilizar técnicas de control que aprovechen la identificación obtenida.

1.3. Presentación

Cuando se controla un robot móvil se desea obtener un comportamiento cercano al indicado, con un margen de error mínimo o de ser posible nulo. Para lograr el comportamiento óptimo, es recomendable conocer la función de transferencia del robot, este trabajo presenta el marco teórico y práctico para estimarla utilizando conceptos de filtrado adaptable y una vez conocida, implantar el controlador proporcional integral y derivativo (PID) para hacer más eficiente el desempeño del robot (figura 1.1) (Ogata).

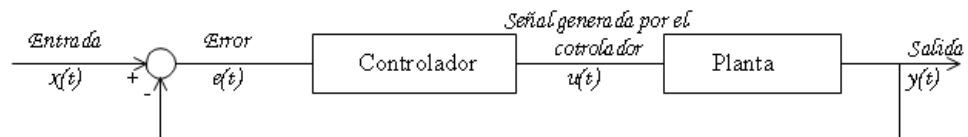


Figura 1.1: Controlador

Los filtros adaptables (Haykin) son usados para ajustar de manera dinámica los coeficientes o parámetros de un sistema con el objetivo de obtener un desempeño favorable ante condiciones de operación variantes en el tiempo; entre las aplicaciones más empleadas se encuentran: la predicción, identificación de sistemas, ecualización y la cancelación de interferencias.

Utilizando la modalidad de identificación de sistemas, será posible estimar el

modelo de un sistema desconocido, en este caso la función de transferencia del robot. A través de un método experimental, en el que se obtiene la respuesta del robot a cierta excitación de entrada, con esta información se realiza el procesamiento de estas señales utilizando los algoritmos que de acuerdo al modelo permitirán conocer los coeficientes de la función.

La figura 1.2 describe la identificación de sistemas (Widrow) usando filtrado adaptable en la forma general y la respuesta del sistema desconocido a una entrada es indicada por la ecuación (1.1):

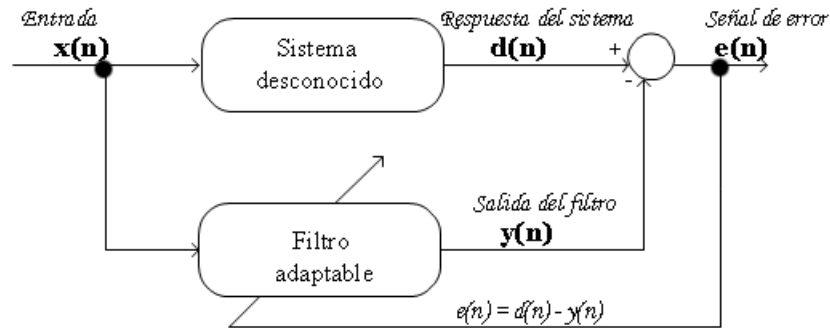


Figura 1.2: Representación de la identificación de sistemas

$$d(n) = \sum_{k=0}^N h_k x(n-k) \quad (1.1)$$

Para el caso de un filtro FIR (Filtro de respuesta a Impulso Finito) la salida del filtro adaptable puede ser representado por:

$$y(n) = \sum_{k=0}^{N-1} W_k x(n-k) \quad (1.2)$$

donde $x(n)$ es la señal de entrada, $d(n)$ la señal deseada, $e(n)$ es la señal de error, $y(n)$ la salida del filtro, h_k y W_k son los coeficientes del sistema desconocido y

del filtro adaptable respectivamente.

En caso de utilizar un filtro IIR (Filtro de Respuesta Impulso Infinito), la salida depende de las entradas presentes y pasadas así como de las salidas previas; por lo que estará en función de dos conjuntos de coeficientes denominados a_k y b_k , como se muestra en la ecuación (1.3).

$$y(n) = \sum_{k=0}^L a_k x(n-k) + \sum_{k=1}^L b_k y(n-k) \quad (1.3)$$

Como se aprecia en la figura 1.2, la señal de entrada $x(n)$ es ingresada tanto al sistema desconocido como al sistema adaptable; el algoritmo de adaptación deberá reducir el error existente entre la señal deseada (respuesta del robot) y la salida del filtro adaptable y al cabo de ciertas iteraciones se obtendrá una convergencia entre los parámetros del sistema original y el estimado por el filtro adaptable, con lo que se tendrá identificado el sistema.

1.4. Objetivos generales

Estudiar, desarrollar e implantar un protocolo para caracterizar un robot móvil, aplicando las técnicas de control más usadas en la industria y en las comunicaciones; encontrar la función de transferencia de un robot móvil, que describe la dinámica y cinemática de este; con la finalidad de mejorar su desempeño.

1.5. Objetivos específicos

- Construir el robot móvil para su modelado
- Estudiar y analizar el marco teórico del filtrado adaptable

- Estudiar y analizar las características de un controlador PID
- Describir las ventajas del algoritmo LMS
- Implantar el algoritmo adecuado a la estructura del modelo del robot
- Realizar pruebas experimentales para obtener resultados útiles para el análisis
- Plantear el procedimiento para la obtención de la función de transferencia

1.6. Metodología

Para cumplir con los objetivos propuestos, se deberán realizar las siguientes actividades:

- Construcción del robot
- Adecuar al robot
- Realizar pruebas con el robot, para encontrar la relación que existirá entre la velocidad y el valor de PWM
- Obtener la velocidad sin realimentación
- Obtener la velocidad con realimentación
- Programación para identificar filtros adaptables
- Programación de algoritmos LMS no recursivos FIR
- Programación de algoritmos LMS recursivos IIR
- Utilizar dispositivos externos para obtención de la velocidad del robot
- Pruebas y analisis de resultados
- Comparación entre algoritmos usados y otras técnicas de identificación de sistemas

1.7. Estructura del trabajo tesis

El trabajo consta de siete capítulos, los cuales describen el marco contextual, la teoría requerida, análisis de los resultados obtenidos y las conclusiones finales. Cada capítulo presenta la información desde lo más general a lo más específico, resaltando los aspectos relevantes durante el desarrollo; además se incluye la introducción y los fundamentos teóricos en cada capítulo.

El capítulo dos tiene la teoría requerida para comprender las ventajas de los filtros adaptables; así como de los algoritmos adaptables más empleados para el cálculo y actualización de los parámetros del filtro.

En el capítulo tres se realiza un análisis del robot, para encontrar de manera analítica la estructura del modelo del robot móvil con la distribución diferencial, podrán ser apreciadas las ventajas de calcular la función de transferencia mediante metodologías experimentales; en este caso el filtrado adaptable.

El capítulo cuatro incluye la información relacionada con la construcción del robot y las distintas etapas que se requieren para tener un dispositivo funcional.

El capítulo cinco presenta las estructuras de los algoritmos adaptables implantados.

El capítulo seis contiene la descripción de las distintas pruebas realizadas y el análisis de los datos obtenidos.

El capítulo siete tiene las conclusiones que se obtuvieron, una vez realizadas las diferentes etapas que consta el trabajo.

Capítulo 2

Filtros adaptables

El objetivo de los filtros adaptables es minimizar el error cuadrático o cualquier función creciente del error, a través de la ejecución de algoritmos adaptables que permitirán ajustar de manera dinámica los coeficientes o parámetros de un sistema determinado, con el objetivo de obtener un desempeño favorable ante condiciones variantes en el tiempo.

Entre las aplicaciones de filtrado adaptable se encuentran: la predicción, la identificación o modelado de sistemas, la ecualización y la cancelación de interferencias.

En general un filtro adaptable tiene dos procesos básicos (figura 2.1):

1. Un proceso de filtrado; en el cuál se obtiene un dato de salida en respuesta a datos de entrada .
2. Un proceso adaptativo; se ajustan los coeficientes del filtro de acuerdo a un algoritmo.

Por lo tanto el objetivo del filtro adaptable es:

Encontrar los coeficientes del filtro, de manera que la salida del filtro $y(n)$ se parezca lo más posible a una señal deseada $d(n)$, de acuerdo a una señal de entrada $x(n)$ que ingresa al filtro.

Para obtener los coeficientes, se calcula la diferencia que existe entre la señal deseada $d(n)$ y la salida del filtro $y(n)$, denominada señal de error $e(n)$ y se modifican los coeficientes del filtro en la dirección deseada.

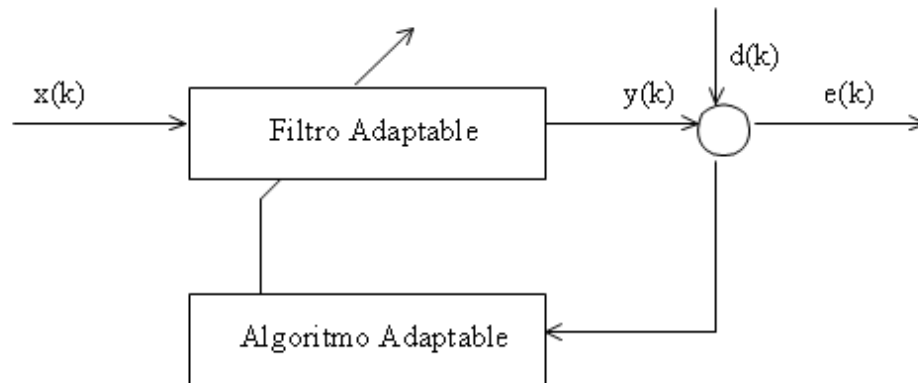


Figura 2.1: Filtro adaptable

La búsqueda de los coeficientes se realiza utilizando el método de máxima pendiente, minimizando el error cuadrático medio calculando el gradiente de la función (derivada de la función con respecto a cada una de sus variables)

De acuerdo a la estructura del modelo se tendrá que seleccionar ya sea un filtro FIR o un filtro IIR para su modelado.

2.1. Filtro de Wiener

Empleando por facilidad un filtro de respuesta finita FIR, el cuál se caracteriza por que la salida depende de las entradas pasadas y la actual; representada por la ecuación (2.1):

$$y(n) = \sum_{k=0}^P b_k x(n-k) \quad (2.1)$$

Por lo tanto la señal de error en un instante determinado es:

$$e(n) = d(n) - y(n) \quad (2.2)$$

Sustituyendo (2.1) en (2.2):

$$e(n) = d(n) - \sum_{k=0}^P b_k x(n-k) \quad (2.3)$$

El método por el cual el filtro se adapta es definir una función J de coste, dependiente del error y que sea más pequeña cuanto más pequeño sea el error e intentar minimizarla (reduciendo el error) modificando los coeficientes del filtro.

Una opción directa es definir el error cuadrático medio como:

$$J = E [e^2(n)] \quad (2.4)$$

donde J es el error instantáneo (elevado al cuadrado para que siempre sea positivo y al minimizarlo se reduzca el error, acercándolo a cero).

Usando el análisis de predicción lineal para encontrar los coeficientes del filtro; definiendo un conjunto de datos de entradas :

$$U_{n-1} = \{u[n-1], u[n-2], \dots, u[n-M]\} \quad (2.5)$$

La predicción de la muestra $u(n)$ dado el conjunto de muestras anteriores, se representa por:

$$U(n|_{u_{n-1}}) = f \{u[n-1], u[n-2], \dots, u[n-M]\} \quad (2.6)$$

Esto es:

$$U(n|_{u_{n-1}}) = \sum_{k=1}^M W_k u(n-k) \quad (2.7)$$

Por lo tanto el error de predicción es :

$$f_m(n) = u(n) - u(n|_{u_{n-1}}) \quad (2.8)$$

Minimizando el error a su valor cuadrático medio:

$$J(w) = E \left\{ |f_m(n)|^2 \right\} \quad (2.9)$$

Si W_k^* es una variable compleja, entonces la ecuación (3.7) se transforma en:

$$u(n | u_{n-1}) = \sum_{k=1}^M W_k^* u(n-k) \quad (2.10)$$

Por lo tanto el error será complejo; usando la expresión (2.9), el error cuadrático medio será:

$$J(w) = E \{ f_m(n) f_m^*(n) \} \quad (2.11)$$

Usando notación matricial, para representar los coeficientes del filtro:

$$W = \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_M \end{pmatrix} \quad (2.12)$$

Así mismo para matriz traspuesta y la matriz hermitiana (conjugado de la matriz traspuesta) de los coeficientes:

$$W^T = \left(w_1 \quad w_2 \quad \cdots \quad w_M \right) \quad (2.13)$$

$$W^H = \left(w_1^* \quad w_2^* \quad \cdots \quad w_M^* \right) \quad (2.14)$$

El vector de entradas esta dado por:

$$u(n-1) = \begin{pmatrix} u(n-1) \\ u(n-2) \\ \vdots \\ u(n-M) \end{pmatrix} \quad (2.15)$$

Usando (2.14) y (2.15) en (2.10):

$$u(n|u_{n-1}) = \begin{pmatrix} w_1^* & w_2^* & \cdots & w_M^* \end{pmatrix} \begin{pmatrix} u(n-1) \\ u(n-2) \\ \vdots \\ u(n-M) \end{pmatrix} \quad (2.16)$$

Se puede representar como:

$$u(n|u_{n-1}) = W^H u(n-1) \quad (2.17)$$

La función de estimación del error y el conjugado de la misma están expresadas por:

$$f_m(n) = u(n) - W^H u(n-1) \quad (2.18)$$

$$f_m^*(n) = u^*(n) - u^H(n-1)W \quad (2.19)$$

De acuerdo a la ecuación (2.9), el error cuadrático medio será:

$$J(W) = E \{f_m(n)f_m^*(n)\} \quad (2.20)$$

Sustituyendo (2.18) y (2.19) en (2.20):

$$J(W) = E \left\{ [u(n) - W^H u(n-1)] [u^*(n) - u^H(n-1)W] \right\} \quad (2.21)$$

Desarrollando (2.21):

$$\begin{aligned} J(W) &= E \{ u(n)u^*(n) \} - W^H E \{ u(n-1)u^*(n) \} E \{ u(n)u^H(n-1) \} W \\ &+ W^H E \{ u(n-1)u^H(n-1) \} W \end{aligned} \quad (2.22)$$

Analizando cada término de esta ecuación:

1. Primer término:

$$E \{ u(n)u^*(n) \} = \sigma_u^2 \quad (2.23)$$

donde σ_u^2 es la varianza

2. Segundo término

$$E \{ u(n-1)u^*(n) \} = E \left\{ \begin{pmatrix} u(n-1) \\ u(n-2) \\ \vdots \\ u(n-M) \end{pmatrix} u^*(n) \right\} = E \left\{ \begin{pmatrix} E \{ u(n-1)u^*(n) \} \\ E \{ u(n-2)u^*(n) \} \\ \vdots \\ E \{ u(n-M)u^*(n) \} \end{pmatrix} \right\} \quad (2.24)$$

Definiendo $E \{ u(n-k)u^*(n-j) \} = r(j-k)$; al sustituir en 2.24 se obtiene el vector de correlación de la entrada.

$$\mathbf{r} = \begin{pmatrix} r(-1) \\ r(-2) \\ \vdots \\ r(-m) \end{pmatrix} \quad (2.25)$$

3. Tercer término

$$E \left\{ u(n)u^H(n-1) \right\} = E \left\{ \begin{pmatrix} u(n)u^*(n-1) & u(n)u^*(n-2) & \cdots & u(n)u^*(n-M) \end{pmatrix} \right\} = \mathbf{r}^H \quad (2.26)$$

4. Cuarto término

$$E \left\{ u(n-1)u^H(n-1) \right\} = E \left\{ \begin{pmatrix} u(n-1) \\ u(n-2) \\ \vdots \\ u(n-M) \end{pmatrix} \begin{pmatrix} u^*(n-1) & u^*(n-2) & \cdots & u^*(n-M) \end{pmatrix} \right\} \quad (2.27)$$

Se obtiene la matriz:

$$E \left\{ \begin{pmatrix} u(n-1)u^*(n-1) & u(n-1)u^*(n-2) & \cdots & u(n-1)u^*(n-M) \\ u(n-2)u^*(n-1) & u(n-2)u^*(n-2) & \cdots & u(n-2)u^*(n-M) \\ \vdots & \vdots & \cdots & \vdots \\ u(n-M)u^*(n-1) & u(n-M)u^*(n-2) & \cdots & u(n-M)u^*(n-M) \end{pmatrix} \right\} \quad (2.28)$$

Aplicando el operador esperanza a cada uno de los términos, se obtiene la matriz de correlación:

$$\mathbf{R} = \begin{pmatrix} r(0) & r(1) & \cdots & r(M-1) \\ r(-1) & r(0) & \cdots & r(M-2) \\ \vdots & \vdots & \cdots & \vdots \\ r(-M+1) & r(-M+2) & \cdots & r(0) \end{pmatrix} \quad (2.29)$$

Sustituyendo en (2.29), (2.28), (2.27) y (2.23) en la ecuación (2.22)

$$J(W) = \sigma_u^2 - \mathbf{W}^H \mathbf{r} - \mathbf{r}^H \mathbf{W} + \mathbf{W}^H \mathbf{R} \mathbf{W} \quad (2.30)$$

El objetivo es encontrar el vector W que minimice el error cuadrático medio; por lo tanto, derivando la ecuación (2.30).

$$\frac{\partial J(W)}{\partial(W)} = 0 \quad (2.31)$$

Una vez más aplicando (2.31) en (2.30):

Si para el segundo término se propone:

$$g = \mathbf{W}^H \mathbf{r} \quad (2.32)$$

El vector W complejo, queda de la siguiente forma:

$$W_K = W_{KReal} + jW_{KImaginaria} \quad (2.33)$$

La ecuación (3.32) será:

$$g = \sum_{K=1}^M W_k^* r_K = \sum_{K=1}^M (W_{KReal} + jW_{KImaginaria}) r_K \quad (2.34)$$

Aplicando (2.33)

$$\frac{\partial J(W)}{\partial(W)} = \begin{pmatrix} \frac{\partial g}{\partial W_{1Real}} + j \frac{\partial g}{\partial W_{1Imag}} \\ \frac{\partial g}{\partial W_{2Real}} + j \frac{\partial g}{\partial W_{2Imag}} \\ \vdots \\ \frac{\partial g}{\partial W_{MReal}} + j \frac{\partial g}{\partial W_{MImag}} \end{pmatrix} \quad (2.35)$$

Se obtiene:

$$\frac{\partial g}{\partial W_{KReal}} = r_k \quad (2.36)$$

$$\frac{\partial g}{\partial W_{KImag}} = -jr_K \quad (2.37)$$

Sustituyendo:

$$\frac{\partial g}{\partial W} = \begin{pmatrix} r_1 + (j)(-j)r_1 \\ r_2 + (j)(-j)r_2 \\ \vdots \\ r_M + (j)(-j)r_M \end{pmatrix} = \begin{pmatrix} 2r_1 \\ 2r_2 \\ \vdots \\ 2r_M \end{pmatrix} = 2\mathbf{r} \quad (2.38)$$

Para el tercer elemento:

$$g = \mathbf{r}^H \mathbf{W} = \sum_{K=1}^M r_k^* (W_{KReal} + jW_{KImag}) \quad (2.39)$$

$$\frac{\partial g}{\partial W} = \begin{pmatrix} r_1^* + (j)(j)r_1^* \\ r_2^* + (j)(j)r_2^* \\ \vdots \\ r_M^* + (j)(j)r_M^* \end{pmatrix} = 0 \quad (2.40)$$

Para el último término:

$$g = \mathbf{W}^H \mathbf{R} \mathbf{W} \quad (2.41)$$

Aplicando el análisis previo:

$$\frac{\partial g}{\partial W} = 2\mathbf{R} \mathbf{W} \quad (2.42)$$

Sustituyendo (2.23), (2.28), (2.40) y (2.42) en la ecuación (2.30):

$$\frac{\partial J(W)}{\partial W} = 0 - 2\mathbf{r} + 0 + 2\mathbf{R} \mathbf{W} = -\mathbf{r} + \mathbf{R} \mathbf{W} \quad (2.43)$$

despejando:

$$\mathbf{R}\mathbf{W} = \mathbf{r} \quad (2.44)$$

Por lo tanto, la expresión para encontrar el filtro óptimo esta dada por:

$$\mathbf{W} = \mathbf{R}^{-1} \mathbf{r} \quad (2.45)$$

la ecuación anterior describe al filtro de Wiener-Hopf.

Donde R es la matriz de correlación o de autocorrelación del vector de entradas $R = E[u(n)u^T(n)]$, de tamaño $(M + 1)(M + 1)$; r es el vector de correlación cruzada entre la entradas y la señal deseada $r = [d(n)u(n)]$ de $(M + 1)$ y W es el vector de coeficientes óptimos que minimizan el error cuadrático medio.

2.2. Algoritmos adaptables

El objetivo es encontrar el filtro óptimo W que satisface las ecuaciones de Wiener-Hopf, ecuación (2.45). La solución a este sistema de ecuaciones puede tener una carga computacional excesiva si el número de coeficientes es elevado. Una forma alternativa a la solución directa es emplear algún procedimiento iterativo de solución. Entre los más usados se encuentran el método de máximo descenso (*SteepestDescentAlgorithm*) y el algoritmo del error cuadrático mínimo o LMS (*LeasMeanSquare*).

2.2.1. Algoritmo de máximo descenso

Uno de los métodos más antiguos es el de máximo descenso; el objetivo es minimizar el error cuadrático medio (J_{min}); procediendo bajo el siguiente algoritmo:

1. Se parte de un vector de coeficientes; cuyo valor inicial es un vector nulo.
2. Calcular el vector gradiente definido como la derivada del error cuadrático medio, $J[(n)]$ con respecto al vector de coeficientes $W(n)$.

3. Calcular el nuevo filtro $W(n + 1)$, cambiando el filtro previo en la dirección contraria a la del vector gradiente.
4. Repetir hasta alcanzar los pesos óptimos

2.2.2. Algoritmo LMS para filtros no recursivos FIR

Los filtros FIR se describen por la ecuación (2.46) y la representación gráfica por medio de la figura 2.2:

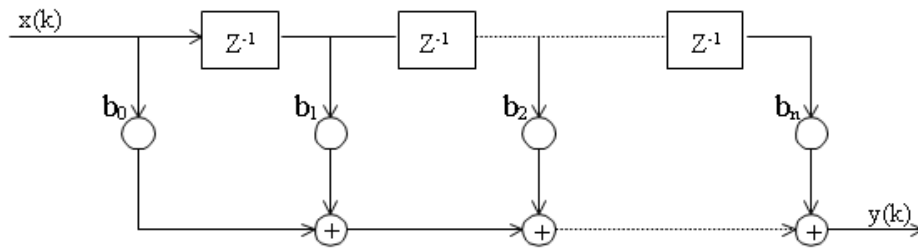


Figura 2.2: Filtro FIR

$$y(n) = \sum_{k=0}^P b_k x(n - k) \quad (2.46)$$

La salida depende únicamente de las entradas actuales y pasadas.

Este tipo de filtros son inherentemente estables, al contar con polos en el origen del círculo unitario del plano Z , además al determinar el error cuadrático medio mínimo se tiene una única solución que representará la mejor solución del filtro.

El error en un instante cualquiera está dado por la diferencia entre la señal deseada o de referencia y la salida del filtro:

$$e(n) = d(n) - y(n) = d(n) - \sum_{k=0}^P b_k x(n - k) \quad (2.47)$$

Para minimizar la señal de error se usa el método de mínimos cuadrados:

$$e(n) = e(n)^2 = \left(d(n) - \sum_{k=0}^P b_k x(n-k) \right)^2 \quad (2.48)$$

El método más usado para reducir una función que depende de varios parámetros es su gradiente (vector que apunta a la dirección de máximo incremento) y moverse en sentido contrario.

$$\nabla J = \left(\frac{\partial J}{\partial b_0}, \frac{\partial J}{\partial b_1}, \dots, \frac{\partial J}{\partial b_P} \right) \quad (2.49)$$

El gradiente de una función es un vector cuyas componentes son las derivadas parciales con respecto a todas sus variables.

El vector actual de coeficientes será:

$$b(n+1) = b(n) - \mu \nabla J \quad (2.50)$$

donde μ es el factor de adaptación durante el proceso y que será en la dirección opuesta al gradiente.

Calculando el gradiente de la ecuación (2.48):

$$\frac{\partial J}{\partial b_K} = \frac{\partial}{\partial b_K} \left(d(n) - \sum_{k=0}^P b_k x(n-k) \right)^2 = 2 \left(d(n) - \sum_{k=0}^P b_k x(n-k) \right) (-x(n-k)) \quad (2.51)$$

Quedando en forma reducida:

$$\frac{\partial J}{\partial b_K} = -2e(n)x(n-k) \quad (2.52)$$

Sustituyendo (2.52) en (2.50) se obtiene la ecuación para actualizar los coeficientes del filtro:

$$b(n+1) = b(n) + 2\mu e(n) x(n-k) \quad (2.53)$$

Para la actualización (figura 2.3) se requiere para cada iteración del algoritmo de:

La salida del filtro:

$$y(n) = \mathbf{b}^T(n)x(n) \quad (2.54)$$

La señal de error:

$$e(n) = d(n) - y(n) \quad (2.55)$$

Los coeficientes del filtro:

$$\mathbf{b}(n) = \left[b_0(n) \quad b_1(n) \quad \cdots \quad b_{M-1}(n) \right]^T \quad (2.56)$$

La señal de entrada:

$$\mathbf{x}(n) = \left[x(n) \quad x(n-1) \quad \cdots \quad x(n-M+1) \right]^T \quad (2.57)$$

2.2.3. Algoritmo LMS para filtros recursivos IIR

Un filtro recursivo tiene polos y ceros en su función de transferencia, la presencia de los polos podrían hacer al sistema inestable al caer estos fuera del círculo unitario y

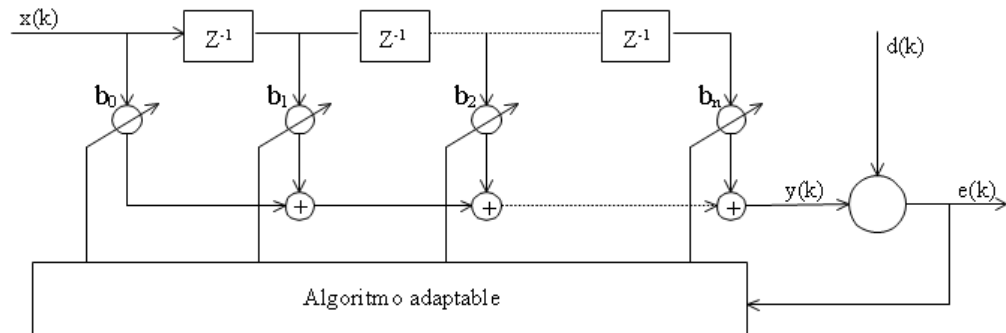


Figura 2.3: Algoritmo LMS para un filtro adaptable FIR

tener mínimos locales. La salida en un instante determinado dependerá tanto de las entradas presentes y pasadas como de las salidas anteriores, se describe con la figura 2.4 y la ecuación (2.58):

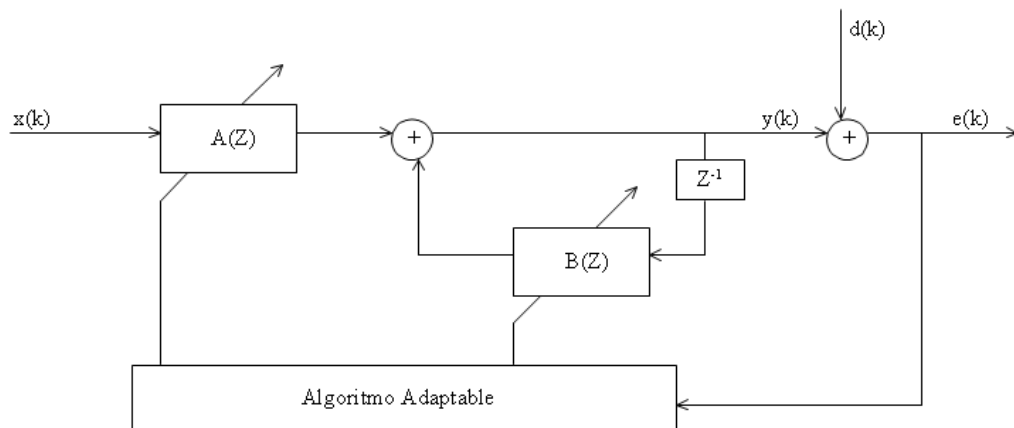


Figura 2.4: Filtro adaptable IIR

$$y_k = \sum_{n=0}^L a_n x_{k-n} + \sum_{n=1}^L b_n x_{k-n} \quad (2.58)$$

Para desarrollar el algoritmo se proponen las siguientes notaciones:

a) Vector de coeficientes

$$W_k = \begin{bmatrix} a_{0k} & a_{1k} & \cdots & a_{Lk} & b_{1k} & \cdots & b_{Lk} \end{bmatrix}^T \quad (2.59)$$

b) Vector de entradas y salidas

$$U_k = \begin{bmatrix} x_k & x_{k-1} & \cdots & x_{k-L} & y_{k-1} & \cdots & y_{k-L} \end{bmatrix}^T \quad (2.60)$$

De la figura 2.4, se tiene que:

$$e_k = d_k - y_k = d_k - W_k^T U_k \quad (2.61)$$

Es muy parecido al caso no recursivo, pero ahora U_k contiene valores tanto de las entradas como de las salidas y W_k tendrá los coeficientes.

Encontrando nuevamente el vector gradiente:

$$\nabla_k = \left(\frac{\partial e^2}{\partial W_k} \right) = 2e \frac{\partial e}{\partial W_k} \quad (2.62)$$

$$\nabla_k = 2e_k \left[\frac{\partial e_k}{\partial a_{0k}} \cdots \frac{\partial e_k}{\partial a_{Lk}} \frac{\partial e_k}{\partial b_{1k}} \cdots \frac{\partial e_k}{\partial b_{Lk}} \right]^T$$

de la ecuación (2.61):

$$\nabla_k = -2e_k \left[\frac{\partial y_k}{\partial a_{0k}} \cdots \frac{\partial y_k}{\partial a_{Lk}} \frac{\partial y_k}{\partial b_{1k}} \cdots \frac{\partial y_k}{\partial b_{Lk}} \right]^T \quad (2.63)$$

El problema que presenta la ecuación (2.63) es que y_k es ahora una función recursiva; definiendo apartir de (2.58); la derivada de la salida con respecto a los coeficientes a_k 's será:

$$\alpha_{nk} \cong \frac{\partial y_k}{\partial a_n} = x_{k-n} + \sum_{l=1}^L b_l \frac{\partial y_{k-l}}{\partial a_n}$$

$$\alpha_{nk} = x_{k-n} + \sum_{l=1}^L b_l \alpha_{n,k-l} \quad (2.64)$$

Ahora la derivada con respectos a los coeficientes b_k 's:

$$\beta_{nk} \cong \frac{\partial y_k}{\partial b_n} = y_{k-n} + \sum_{l=1}^L b_l \frac{\partial y_{k-l}}{\partial b_n}$$

$$\beta_{nk} = y_{k-n} + \sum_{l=1}^L b_l \beta_{n,k-l} \quad (2.65)$$

El gradiente expresado por (2.63) será:

$$\nabla_k = -2e_k \left[\alpha_{0k} \quad \alpha_{1k} \quad \cdots \quad \alpha_{Lk} \quad \beta_{1k} \quad \cdots \quad \beta_{LK} \right]^T \quad (2.66)$$

De manera similar al algoritmo LMS no recursivo:

$$W_{k+1} = W_k - \mu_{a,b} \nabla_k \quad (2.67)$$

Por lo que la actualización de los coeficiente a_k 's y b_k 's se centrará en la obtención de los α_k 's y β_k 's.

El algoritmo LMS para filtros recursivos IIR requiere de:

a) Actualización de los coeficientes a_k 's y b_k 's de la iteración previa, integrados en el vector W_k 's:

$$W_{k+1} = W_k + 2 \mu_{a,b} e_k \left[\alpha_{0k} \quad \alpha_{1k} \quad \cdots \quad \alpha_{Lk} \quad \beta_{1k} \quad \cdots \quad \beta_{LK} \right]^T \quad (2.68)$$

b) Los factores de actualización μ_a y μ_b

c) La señal de error dada por:

$$e_k = d_k - y_k \quad (2.69)$$

donde:

$$y_k = d_k - W_k^T U_k$$

d) La estimación de la derivada de la salida con respecto a las a_k 's y b_k 's; encontradas a través de (2.64) y (2.65).

2.3. Aplicaciones de filtrado adaptable

El objetivo final del filtrado adaptable es minimizar el error cuadrático medio o cualquier función creciente del error; a continuación se describen las estructuras de las posibles aplicaciones.

2.3.1. Predicción

Esta estructura realiza la predicción del futuro de la señal en un tiempo determinado, en función de las muestras anteriores de la señal de entrada.

La figura 2.6 representa este caso; en la cuál la salida del filtro utiliza las muestras anteriores de la señal $x(n)$ para encontrar la salida $y(n)$; el objetivo es que $y(n)$ sea igual a $x(n)$.

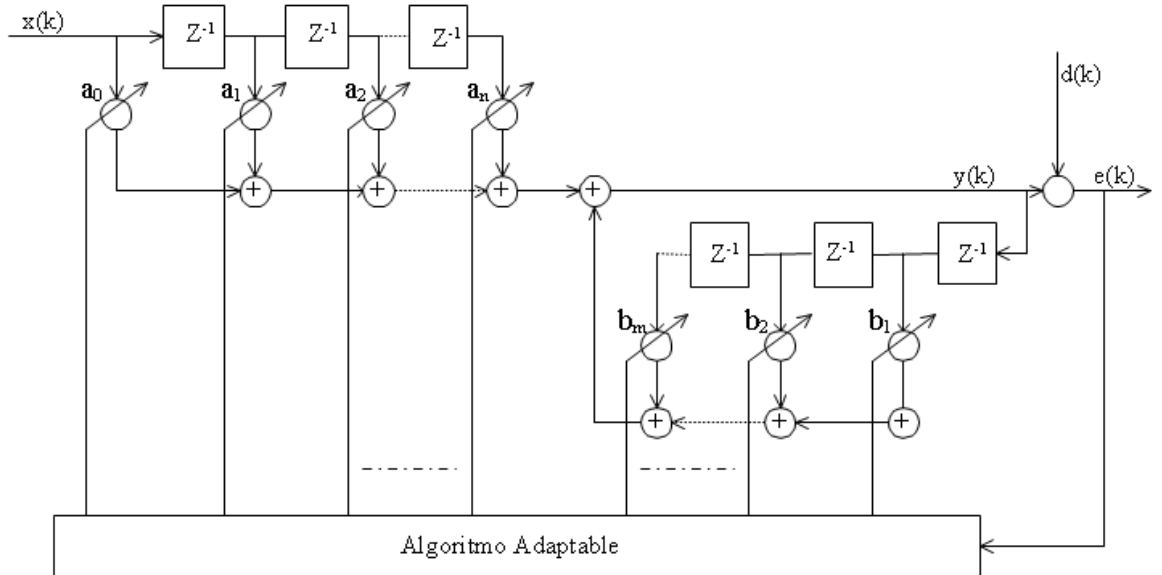


Figura 2.5: Algoritmo LMS filtros IIR

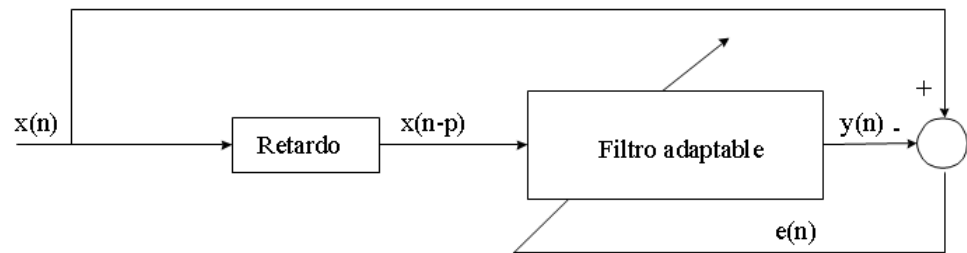


Figura 2.6: Predictor

2.3.2. Identificación de sistemas

En este caso se busca que la salida del filtro adaptable sea igual a la salida del sistema desconocido. Para lo cuál los dos sistemas tienen la misma entrada, en el momento que las salidas de ambos sean iguales, entonces se tendrá identificado al sistema desconocido; por lo tanto se puede asumir que la función de transferencia del sistema desconocido es la misma del filtro adaptable para las mismas entradas.

La figura 2.7 muestra la representación gráfica de esta aplicación.

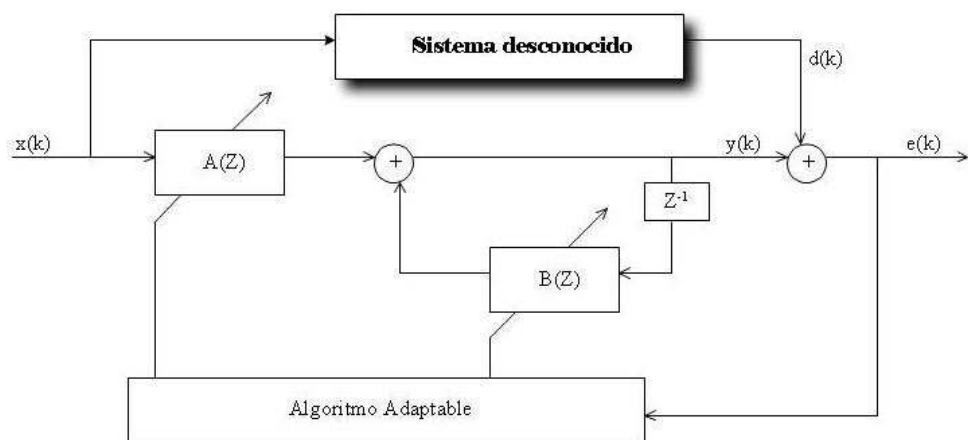


Figura 2.7: Identificación de sistemas

2.3.3. Ecualización

Se considera minimizado el error cuando la salida del filtro adaptable $y(n)$ es igual a la señal deseada, en este caso a la entrada $x(n)$. El objetivo del filtro es deshacer lo que hace el sistema desconocido. En teoría de sistemas esta acción es equivalente a decir que el filtro adaptable tiene como función de transferencia la inversa del sistema desconocido; la figura 2.8 muestra el esquema de esta estructura.

2.3.4. Cancelación de interferencias

La figura 2.9 presenta la estructura de esta aplicación.

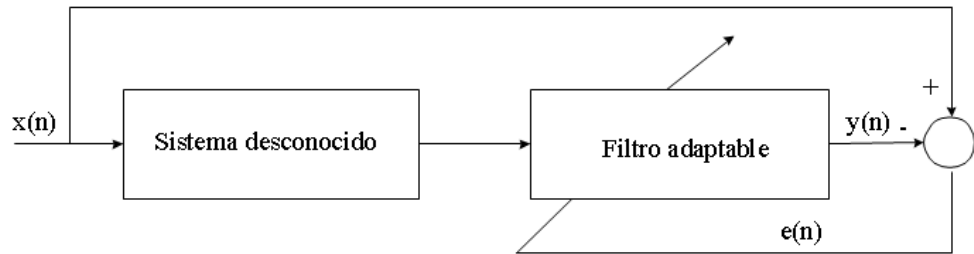


Figura 2.8: Ecuador

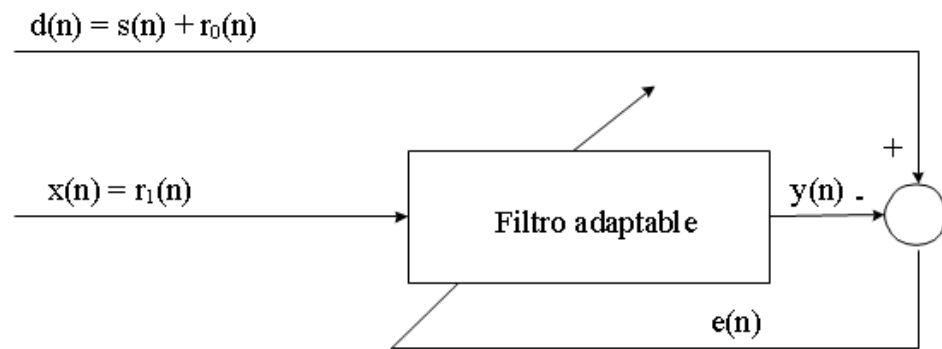


Figura 2.9: Cancelador de interferencias

Donde $s(n)$ es la señal deseada, r_0 es un ruido que afectará a la señal y r_1 es un ruido correlacionado con el anterior; entonces el error cuadrático en el instante n cometido por el filtro adaptable será:

$$e^2 = (d(n) - y(n))^2 = (s(n) + r_0(n) - f(r_1(n)))^2$$

donde $f(r_1)$ es la salida del filtro adaptable.

Debido a la nula correlación entre $s(n)$ y $r_1(n)$ se considera que el sistema modela a la señal de ruido $r_0(n)$ una vez que ha sido minimizado el error cuadrático medio. En este caso, la señal de error $s(n)$, que es la señal que se desea obtener.

Capítulo 3

Modelado del robot

Este capítulo presenta el análisis para encontrar de manera analítica la función de transferencia del robot.

3.1. Dinámica del robot

En la dinámica se consideran las fuerzas que producen el movimiento del robot, se deben calcular las ecuaciones para describir las funciones tanto para la velocidad lineal como la velocidad angular; en este caso se requiere conocer las características de todos los elementos que forman la estructura del robot; a continuación se propone el cálculo realizado.

3.1.1. Movimiento de las llantas

Los movimientos de traslación y rotación del robot móvil se consiguen a través de señales de control que llegan a motores de corriente directa a los cuales se empotran llantas; el modelo de estudio tiene una distribución diferencial que muestra la figura 3.1, compuesto de dos motores que son controlados de manera independiente; cada uno con su respectiva llanta.

Las características a considerar en el análisis de las llantas son:

$\ddot{\theta}_w$ = Aceleración rotacional de la llanta

J_w = Momento de inercia de la llanta

m_w = Masa de la llanta

F_R = Fuerza de fricción

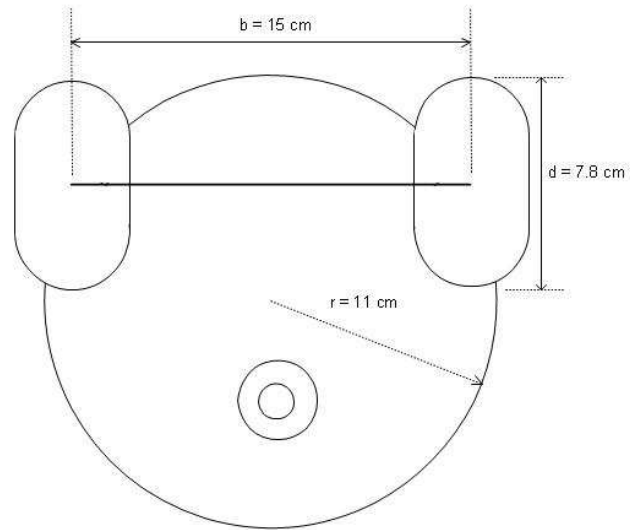


Figura 3.1: Robot móvil diferencial

- R_w = Radio de la llanta
- B_w = Constante de fricción
- τ_w = Torque de la llanta
- F_w = Fuerza de la llanta

La fuerza de fricción de la llanta se obtiene con la relación que existe entre la aceleración rotacional y la constante de fricción de la llanta:

$$F_R = B_w \ddot{\theta}_w \quad (3.1)$$

El torque de la llanta se describe por la ecuación:

$$\tau_w = J_w \ddot{\theta}_w + B_w \dot{\theta}_w \quad (3.2)$$

También el torque de la llanta depende de la fuerza y el radio de esta:

$$\tau_w = F_w R_w \quad (3.3)$$

De (3.3) y sustituyendo en (3.2):

$$F_w = \frac{\tau_w}{R_w} = \frac{1}{R_w} (J_w \ddot{\theta}_w + B_w \dot{\theta}_w) \quad (3.4)$$

3.1.2. Movimiento lineal del robot

El robot se desplaza de manera lineal tanto para adelante como para atrás, para no tener cambios en su trayectoria se debe mantener la misma velocidad en los motores.

Si m_R es la masa del robot y \ddot{X}_R es su aceleración ; entonces la fuerza del robot está dada por:

$$F = m_R \ddot{X}_R \quad (3.5)$$

Considerando un robot con dos llantas (figura 3.1), la fuerza de ambas llantas proporcionarán la fuerza del robot.

$$F = F_{WI} + F_{WD} = m_R \ddot{X}_R = m_R \dot{v}_R \quad (3.6)$$

Sustituyendo (3.4) en (3.6):

$$F = \frac{1}{R_{wI}} (J_{wI} \ddot{\theta}_{wI} + B_{wI} \dot{\theta}_{wI}) + \frac{1}{R_{wD}} (J_{wD} \ddot{\theta}_{wD} + B_{wD} \dot{\theta}_{wD}) \quad (3.7)$$

Calculando la transformada de Laplace de (3.7) y combinando con (3.6) para obtener la velocidad del robot.

$$V_R(S) = W_{wI}(S) \left(\frac{J_{wI}S + B_{wI}}{R_{wI}m_R S} \right) + W_{wD}(S) \left(\frac{J_{wD}S + B_{wD}}{R_{wD}m_R S} \right) \quad (3.8)$$

Representando esta ecuación en forma matricial:

$$V_R(S) = \begin{pmatrix} \frac{J_{wI}S+B_{wI}}{R_{wI}m_R S} & \frac{J_{wD}S+B_{wD}}{R_{wD}m_R S} \end{pmatrix} \begin{pmatrix} W_{wI}(S) \\ W_{wD}(S) \end{pmatrix} \quad (3.9)$$

3.1.3. Movimiento rotacional del robot

El robot deberá cambiar de dirección de desplazamiento o rotar sobre su mismo eje un cierto ángulo; para lo cual se deben enviar a los motores valores diferentes de velocidad.

Las convenciones a usar son:

$\ddot{\theta}$ = Aceleración rotacional del robot

J_R = Momento de inercia del robot

R_R = Radio del robot

τ_R = Torque del robot

B_R = Constante de fricción del robot

De manera similar al movimiento lineal, se emplea la ecuación (3.2); para describir el torque del robot:

$$\tau_R = J_R \ddot{\theta}_R + B_R \dot{\theta}_R \quad (3.10)$$

La fuerza del robot estará dada por la diferencia de las fuerzas existente entre las dos llantas, sustituyendo (3.3) y (3.7):

$$\tau_R = (F_{wI} - F_{wD}) R_R$$

$$\tau_R = \left(\frac{J_{wI}\ddot{\theta}_{wI} + B_{wI}\dot{\theta}_{wI}}{R_{wI}} \right) R_R - \left(\frac{J_{wD}\ddot{\theta}_{wD} + B_{wD}\dot{\theta}_{wD}}{R_{wD}} \right) R_R \quad (3.11)$$

Sustituyendo (3.11) en (3.10) y al obtener la transformada de laplace se llega a:

$$W_R(S) = \frac{W_{wI}(S) R_R}{R_{wI}} \left(\frac{J_{wI}S + B_{wI}}{J_R S + B_R} \right) - \frac{W_{wD}(S) R_R}{R_{wD}} \left(\frac{J_{wD}S + B_{wD}}{J_R S + B_R} \right) \quad (3.12)$$

Representando ahora esta ecuación en forma matricial:

$$W_R(S) = \left(\begin{array}{cc} \frac{R_R}{R_{wI}} \left(\frac{J_{wI}S + B_{wI}}{J_R S + B_R} \right) & - \frac{R_R}{R_{wD}} \left(\frac{J_{wD}S + B_{wD}}{J_R S + B_R} \right) \end{array} \right) \begin{pmatrix} W_{wI}(S) \\ W_{wD}(S) \end{pmatrix} \quad (3.13)$$

El sistema de ecuaciones para la velocidad lineal y la angular, quedará como:

$$\begin{pmatrix} V_R(S) \\ W_R(S) \end{pmatrix} = \begin{pmatrix} \frac{J_{wI}S + B_{wI}}{R_{wI}m_R S} & \frac{J_{wD}S + B_{wD}}{R_{wD}m_R S} \\ \frac{R_R}{R_{wI}} \left(\frac{J_{wI}S + B_{wI}}{J_R S + B_R} \right) & - \frac{R_R}{R_{wD}} \left(\frac{J_{wD}S + B_{wD}}{J_R S + B_R} \right) \end{pmatrix} \begin{pmatrix} W_{wI}(S) \\ W_{wD}(S) \end{pmatrix} \quad (3.14)$$

Para facilitar su manejo, se define la matriz A como:

$$A = \begin{pmatrix} \frac{J_{wI}S + B_{wI}}{R_{wI}m_R S} & \frac{J_{wD}S + B_{wD}}{R_{wD}m_R S} \\ \frac{R_R}{R_{wI}} \left(\frac{J_{wI}S + B_{wI}}{J_R S + B_R} \right) & - \frac{R_R}{R_{wD}} \left(\frac{J_{wD}S + B_{wD}}{J_R S + B_R} \right) \end{pmatrix} \quad (3.15)$$

El sistema de ecuaciones es ahora:

$$\begin{pmatrix} V_R(S) \\ W_R(S) \end{pmatrix} = A \begin{pmatrix} W_{wI}(S) \\ W_{wD}(S) \end{pmatrix} \quad (3.16)$$

Despejando para encontrar las velocidades angulares:

$$\begin{pmatrix} W_{wI}(S) \\ W_{wD}(S) \end{pmatrix} = A^{-1} \begin{pmatrix} V_R(S) \\ W_R(S) \end{pmatrix} \quad (3.17)$$

La matriz inversa de (3.15) es:

$$A^{-1} = \begin{pmatrix} \frac{m_R R_{wI} S}{2(J_{wI} S + B_{wI})} & \frac{R_{wI}}{2R_R} \left(\frac{J_R S + B_R}{J_{wI} S + B_{wI}} \right) \\ \frac{m_R R_{wI} S}{2(J_{wD} S + B_{wD})} & -\frac{R_{wD}}{2R_R} \left(\frac{J_R S + B_R}{J_{wD} S + B_{wD}} \right) \end{pmatrix} \quad (3.18)$$

Sustituyendo (3.18) en (3.17):

$$\begin{pmatrix} W_{wI}(S) \\ W_{wD}(S) \end{pmatrix} = \begin{pmatrix} \frac{m_R R_{wI} S}{2(J_{wI} S + B_{wI})} & \frac{R_{wI}}{2R_R} \left(\frac{J_R S + B_R}{J_{wI} S + B_{wI}} \right) \\ \frac{m_R R_{wI} S}{2(J_{wD} S + B_{wD})} & -\frac{R_{wD}}{2R_R} \left(\frac{J_R S + B_R}{J_{wD} S + B_{wD}} \right) \end{pmatrix} \begin{pmatrix} V_R(S) \\ W_R(S) \end{pmatrix} \quad (3.19)$$

Por lo tanto, la velocidad angular de la llanta izquierda será:

$$W_{wI}(S) = \frac{m_R R_{wI} S}{2(J_{wI} S + B_{wI})} V_R(S) + \frac{R_{wI}}{2R_R} \left(\frac{J_R S + B_R}{J_{wI} S + B_{wI}} \right) W_R(S) \quad (3.20)$$

y la velocidad angular de la llanta derecha:

$$W_{wD}(S) = \frac{m_R R_{wI} S}{2(J_{wD} S + B_{wD})} V_R(S) - \frac{R_{wD}}{2R_R} \left(\frac{J_R S + B_R}{J_{wD} S + B_{wD}} \right) W_R(S) \quad (3.21)$$

Desarrollando las ecuaciones anteriores:

$$W_{wI}(S) 2R_R(J_{wI}S + B_{wI}) = R_R m_R R_{wI} S V_R(S) + R_{wI}(J_R S + B_R) W_R(S) \quad (3.22)$$

$$W_{wD}(S) 2R_R(J_{wD}S + B_{wD}) = R_R m_R R_{wD} S V_R(S) + R_{wD}(J_R S + B_R) W_R(S) \quad (3.23)$$

Por facilidad de manejo, se realizan las siguientes sustituciones:

$$\begin{aligned} A &= 2R_R J_{wI} & F &= 2R_R J_{wD} \\ B &= 2R_R B_{wI} & G &= 2R_R B_{wD} \\ C &= R_R m_R R_{wI} & H &= R_R m_R R_{wD} \\ D &= R_{wI} J_R & I &= R_{wD} J_R \\ E &= R_{wI} B_R & J &= R_{wD} B_R \end{aligned} \quad (3.24)$$

Las ecuaciones quedarán como:

$$W_{wI}(S) (A S + B) = C S V_R(S) + D S W_R(S) + E W_R(S) \quad (3.25)$$

$$W_{wD}(S) (F S + G) = H S V_R(S) + I S W_R(S) + J W_R(S) \quad (3.26)$$

Para encontrar la función de transferencia en el dominio discreto, se usa la siguiente transformación:

$$S = \frac{2}{T} \left(\frac{Z-1}{Z+1} \right) \quad (3.27)$$

sustituyendo (3.27) en (3.25) y al desarrollar se obtiene:

$$W_{wI}(Z) (2A - 2AZ^{-1} + BT + BTZ^{-1}) = 2CV_R - 2CZ^{-1}V_R(Z) + 2DW_R(Z) - 2DZ^{-1}W_R(Z) + ETW_R(Z) + ETZ^{-1}W_R(Z)$$

Representando en tiempo discreto:

$$W_{wI}[n](2A + BT) + (-2A + BT)W_{wI}[n - 1] = 2CV_R[n] - 2CV_R[n - 1] + 2DW_R[n] - 2DW_R[n - 1] + ETW_R[n] + ETW_R[n - 1]$$

Despejando la última expresión se encuentra la ecuación para conocer la velocidad angular de la llanta izquierda:

$$W_{wI}[n] = \frac{1}{2A+BT} \left(2CV_R[n] - 2CV_R[n - 1] + 2DW_R[n] - 2DW_R[n - 1] \right) + \frac{1}{2A+BT} \left(ETW_R[n] + ETW_R[n - 1] - (-2A + BT)W_{wI}[n - 1] \right) \quad (3.28)$$

De manera similar, la ecuación para el cálculo de la velocidad angular de la llanta derecha será:

$$W_{wD}[n] = \frac{1}{2F+GT} \left(2HV_R[n] - 2HV_R[n - 1] + 2IW_R[n] - 2IW_R[n - 1] \right) + \frac{1}{2F+GT} \left(JTW_R[n] + JTW_R[n - 1] - (-2F + GT)W_{wD}[n - 1] \right) \quad (3.29)$$

Las ecuaciones (3.28) y (3.29) describen la función de transferencia del robot; por lo que el objetivo se centra en calcular todos los parámetros (A, B, C, D, E, F, G, H, I y J) que las ecuaciones incluyen; como es de notar también se requiere de conocer muestras pasadas y presentes de las velocidades angulares del robot para la obtención de estas en un momento determinado.

Como se ha mencionado con anterioridad, conocer los parámetros resulta una tarea muy complicada ya que los fabricantes no proporcionan los datos necesarios para el cálculo matemático, aún teniendo algunos de ellos, el sistema se vuelve más complejo por todos los componentes que consta la estructura y elementos del robot, así como de

las no linealidades de ellos por lo que se proponen las técnicas de filtrado adaptable para hacer la estimación de una función de transferencia que reproduzca el comportamiento del robot y con ello no sería necesario contar con la información por parte de los fabricantes.

El diagrama general para la estimación de la función de transferencia quedará representado en la figura 3.2:

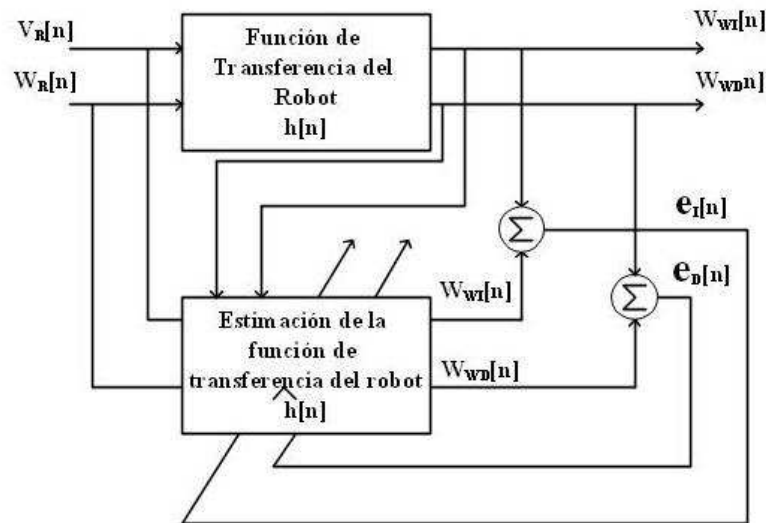


Figura 3.2: Identificación general del robot

3.2. Cinemática del robot

En el análisis del modelo cinemático, se estudia el movimiento del robot, sin considerar las fuerzas que lo producen. El robot consta de dos motores de corriente directa controlados de manera independiente, la figura 3.1 muestra su esquema, estableciendo las siguientes convenciones:

r = Radio de la llanta

d = Distancia entre las llantas

$ticks_{xrev}$ = Número de ticks del encoder para una revolución de la llanta

$ticks_R$ = Número de ticks detectados del encoder derecho

$ticks_L$ = Número de ticks detectados del encoder izquierdo

El desplazamiento de la llanta depende del desarrollo de ésta ($2\pi r$), se puede conocer la distancia recorrida al tomar la lectura de los encoders, así como de la resolución (avance realizado) que se tiene por cada pulso.

La distancia recorrida por la llanta izquierda esta dada por:

$$S_L = \frac{2 \pi ticks_L}{ticks_{xrev}} \quad (3.30)$$

así mismo para el lado derecho:

$$S_R = \frac{2 \pi ticks_D}{ticks_{xrev}} \quad (3.31)$$

Para conocer el desplazamiento del robot, se calcúla el promedio de ambas llanta:

$$S_{Robot} = \frac{S_L + S_D}{2} \quad (3.32)$$

si $S_L = S_D$ el robot se deslaza en línea recta.

La figura 3.3, describe el comportamiento de un robot que avanza y gira al mismo tiempo:

$$S_R = \varphi \left(C + \frac{d}{2} \right) \quad (3.33)$$

para el lado derecho; mientras que:

$$S_L = \varphi \left(C - \frac{d}{2} \right) \quad (3.34)$$

para el lado izquierdo.

Restando (3.34) de (3.33):

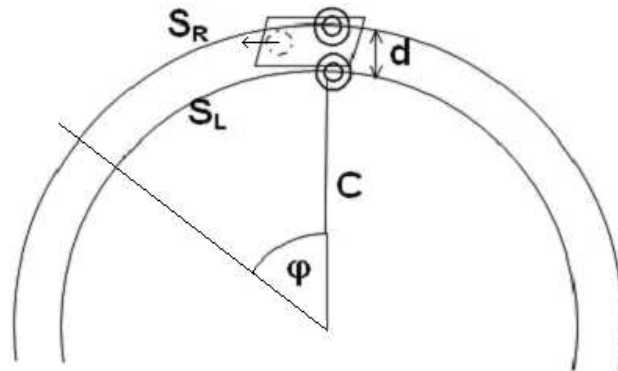


Figura 3.3: Trayectoria del robot

$$S_R - S_L = \varphi d \quad (3.35)$$

Por lo tanto el ángulo de desplazamiento será:

$$\varphi = \frac{S_R - S_L}{d} \quad (3.36)$$

Realizando un análisis equivalente, pero ahora considerando las velocidades:

$$V_R = 2 \pi r \dot{\theta}_R = 2 \pi \omega_R \quad (3.37)$$

$$V_L = 2 \pi r \dot{\theta}_L = 2 \pi \omega_L \quad (3.38)$$

Sustrayendo (3.38) de (3.37) la velocidad de rotación del robot:

$$\omega = \frac{2\pi\omega_R - 2\pi\omega_L}{d} \quad (3.39)$$

Por lo tanto, las ecuaciones de velocidad lineal y angular son:

$$\begin{pmatrix} V \\ \omega \end{pmatrix} = 2\pi r \begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ -\frac{1}{d} & \frac{1}{d} \end{pmatrix} \begin{pmatrix} \omega_L \\ \omega_R \end{pmatrix} \quad (3.40)$$

donde V es la velocidad lineal del robot, ω es la velocidad de rotación del robot, ω_L y ω_R son las velocidades angulares en rev/min de las llantas izquierda y derecha respectivamente.

Después de haber despejado y calculado la inversa; las velocidad angular será para cada llanta:

$$\begin{pmatrix} \omega_L \\ \omega_R \end{pmatrix} = \frac{1}{2\pi r} \begin{pmatrix} 1 & -\frac{d}{2} \\ 1 & \frac{d}{2} \end{pmatrix} \begin{pmatrix} V \\ \omega \end{pmatrix} \quad (3.41)$$

Capítulo 4

Robot móvil

Una vez planteado el marco teórico de este trabajo y al haber encontrado analíticamente las funciones de transferencia en el capítulo anterior, se presentará el diseño y construcción del robot móvil. El diagrama general de la estructura de un robot se muestra en la figura 4.1 y la apariencia física general se muestra en la figura 4.2

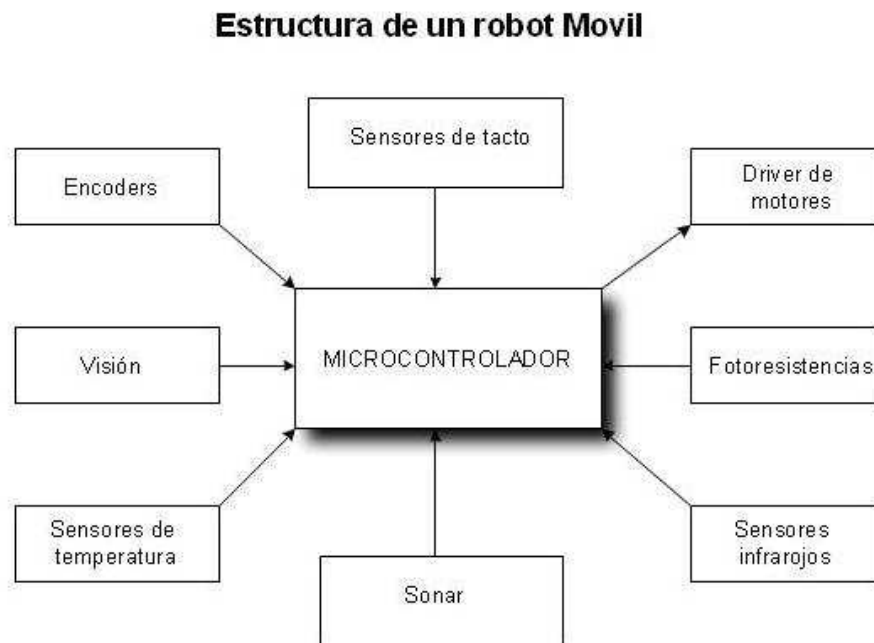


Figura 4.1: Robot móvil

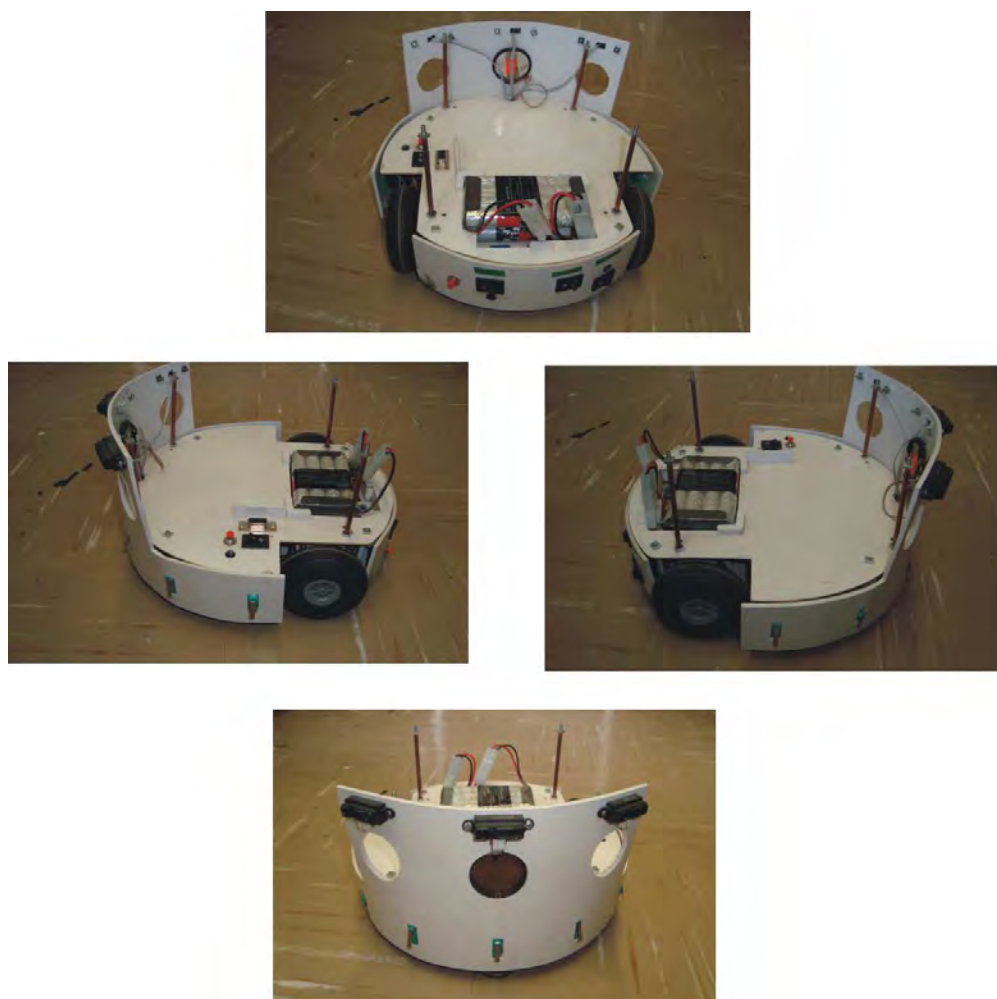


Figura 4.2: Robot móvil visto desde sus cuatro lados

4.1. Descripción del robot

El robot tiene la distribución de motores conocida como "diferencial"; los movimientos son generados a través de dos motores de corriente directa controlados de forma independiente y colocados sobre un mismo eje, a los cuales se les empotran sus respectivas llantas; un tercer apoyo es colocado con la idea de mantener la estabilidad y un contacto uniforme con la superficie de desplazamiento.

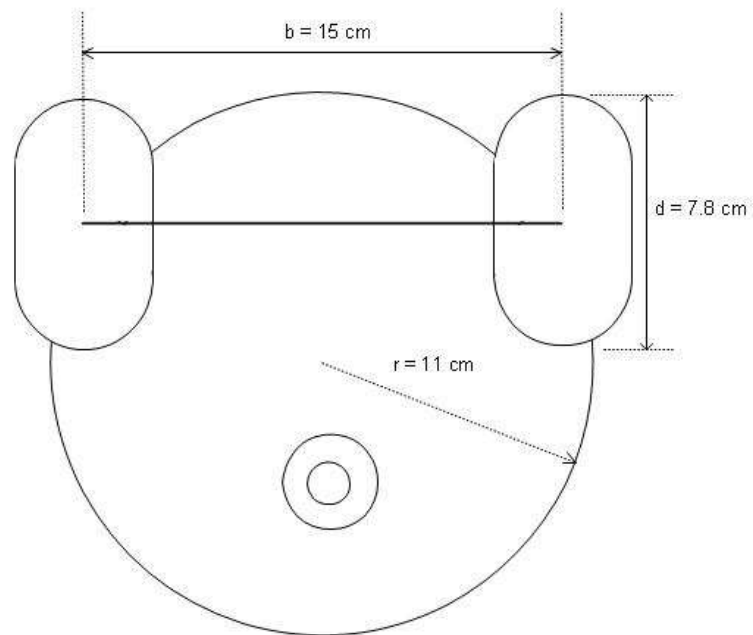


Figura 4.3: Robot móvil distribución diferencial

Esta distribución permite libertad de movimientos; ya sea en línea recta, moverse con un ángulo de desplazamiento determinado o incluso realizar rotaciones sobre su mismo eje.

Los elementos que constituyen al robot son:

- Motores

- Microcontrolador
- Sensores
- Llantas
- Estructura mecánica
- Fuentes de alimentación (pilas)
- Software

4.2. Motores

Se utiliza un sistema compuesto con dos motores de corriente directa, cada uno de ellos contiene un sistema planetario reductor y un encoder integrado. El sistema reductor permite dar mayor potencia al motor, por lo que tendrá la fuerza para cargar y desplazar al robot; mientras que el encoder servirá para conocer la velocidad de rotación de los motores así como la velocidad del robot, el ángulo, la distancia que recorre y en general para implantar técnicas de control.

El motor empleado se muestra en la figura 4.4.



Figura 4.4: Motor Maxon

Características:

a) Motor

- Tensión nominal 6 Volts
- Velocidad al vacío 6840 rpm
- Corriente al vacío 26.7 mA
- Par nominal 6.38 mNm
- Corriente nominal 0.761 A
- Par de arranque 16 mNm
- Corriente de arranque 1.88 A
- Resistencia en bornes 3.20 Ω
- Inductancia en bornes 0.22 mH
- Inercia del rotor 4.30 gcm²
- 22 milímetros de diámetro

b) Sistema planetario

- Sistema reductor 19.1:1

c) Encoder

- Dos canales
- 16 pulsos por revolución al motor con reductor
- 305 pulsos totales por revolución

4.3. Estructura física

Datos importantes en la estructura del robot:

- Radio de la llanta $r = 3.81$ cm.
- 305 pulsos por revolución de la llanta

El desarrollo de la rueda es:

$$\phi_w = \pi d = \pi 7,62 \text{ cm} = 23,93\text{cm} \quad (4.1)$$

La distancia recorrida por pulso:

$$C_m = \frac{\phi_w}{\rho_w}$$

donde ρ_w son los pulsos por revolución

Sustituyendo datos:

$$C_m = \frac{23,93\text{cm}}{305\text{pulsos}} = 0,0784 \frac{\text{cm}}{\text{pulsos}} \quad (4.2)$$

La distancia recorrida por cada llanta estará dada por la relación entre la distancia recorrida por cada pulso y el total de pulsos leídos en un instante dado.

$$\text{Distancia} = \frac{C_m}{\text{Pulsos}} \quad (4.3)$$

Conocida la distancia cubierta se podrá calcular la velocidad a la que gira o desplaza el robot mediante:

$$\text{Velocidad} = \frac{D1 - D2}{T} \quad (4.4)$$

donde $D1$ y $D2$ son la distancia inicial y final recorrida por el robot y T el tiempo de muestreo.

4.4. Microcontrolador

Este dispositivo como su nombre lo indica, servirá para controlar todas las acciones del robot; generando señales para producir el movimiento en cualquier sentido (avanzar, retroceder o girar). Las acciones de control se realizarán por medio de la ejecución de algoritmos programados actuando de acuerdo a señales que ingresan al microcontrolador a través de entradas que provienen de sensores.

Las principales recursos deseables en un microcontrolador son:

Memoria de datos y de programa; para el almacenamiento de datos generados o adquiridos por el programa principal; por lo regular la memoria de programa es no volátil de tipo Flash y la memoria de datos puede ser ya sea volátil (RAM) o no volátil (EEPROM).

Puertos Paralelos; recabará la información que proviene de los distintos sensores conectados al robot y generar señales de control para comandar los movimientos de traslación y rotación del robot.

Convertidor Analógico Digital; como su nombre lo dice convierte señales provenientes de algún transductor analógico a una señal digital, el resultado de la conversión será más sencillo de procesar.

Puerto Serie Asíncrono (SCI); transmite datos en formato serie mediante un protocolo específico, por lo regular es un bit de inicio, 8 bits de datos y un bit de paro; la razón de reconocimiento está dada por la velocidad de transmisión en bauds, es útil para transmitir datos o comandos desde el microcontrolador a una computadora o viceversa.

Puerto Serie Síncrono (SPI); transmite datos serie sincronizados con una fuente de reloj común entre un dispositivo fuente y uno destino; por lo regular se utilizan tres líneas para transferencia de información (Data in, data out y un reloj), el dato transmitido es reconocido cuando se presenta el flanco de reloj (subida o bajada) que ha sido configurado.

Funciones de Temporizador; procesa señales en tiempo real, a través del uso de varias funciones internas como son la captura de pulsos, comparación de salida o generación de un sobreflujo del temporizador.

Generador de Modulación de Ancho de Pulso (PWM); función empleada para generar señales con período fijo y ciclo de trabajo variable, es útil para controlar la velocidad de rotación en motores de cualquier tipo.

Controlador de Interrupciones; este módulo permitirá o cancelará las peticiones de interrupción de un evento interno o externo al procesador. Al reconocer la interrupción el procesador dejará pendiente la ejecución normal del programa para ceder el control a la rutina de interrupción; una vez atendida continuará con la ejecución del programa. La gran mayoría de los periféricos descritos anteriormente permiten ser controlados a través de interrupciones (dependiente de la versión de microcontrolador a emplear).

El dispositivo empleado en este trabajo es el Microcontrolador PIC16F877 de ocho bits de la empresa Microchip, el circuito se muestra en la figura 4.5 ; algunas de sus características generales son:



Figura 4.5: Microcontrolador PIC16F877

- Procesador RISC con arquitectura Harvard
- Frecuencia máxima de operación óptima de 20 MHz
- 8 KBytes de memoria de programa (Flash)
- 368 Bytes de memoria de datos (RAM)

- 255 Bytes de memoria de datos (EEPROM)
- 14 Fuentes de interrupción
- 5 Puertos paralelos (A, B, C, D, E)
- 8 Canales para conversión analógico-digital de 8 o 10 bits de resolución
- Comunicación serie asíncrona (SCI)
- Comunicación serie síncrona (SPI, I2C)
- Temporizador 0 (temporizador/contador) de 8 bits
- Temporizador 1 (temporizador/contador) de 16 bits
- Temporizador 2 (temporizador) de 8 bits
- Módulos CCP1 y CCP2 con funciones de captura, comparación y generadores de PWM
- Permite programación en ensamblador, basic y C entre otros.

Existen varias opciones de programar las versiones de microcontroladores PIC's, una de ellas es utilizar un bootloader que previamente es cargado. Este programa permitirá la programación a través del puerto serie asíncrono; la figura 5.7 muestra la comunicación y la figura 4.8 presenta la fotografía del alambrado de este sistema.

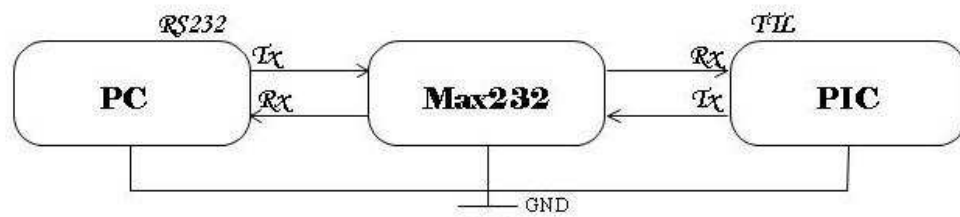


Figura 4.6: Comunicación serie entre la PC y un microcontrolador

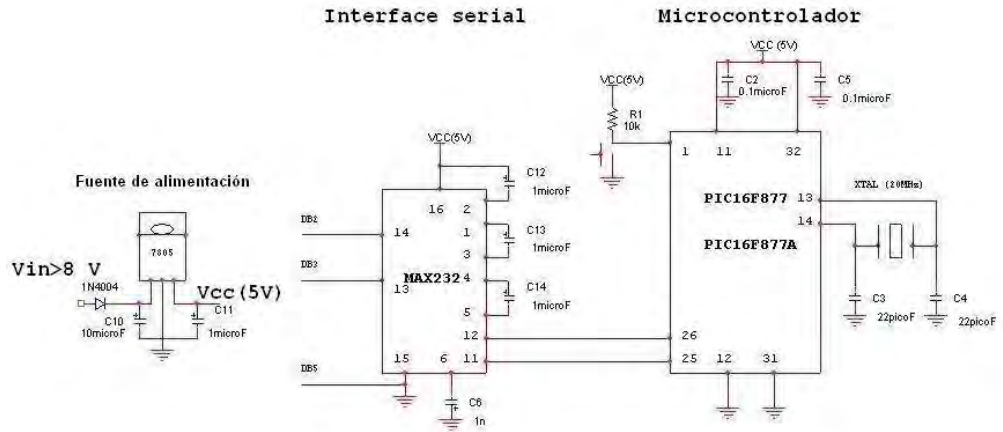


Figura 4.7: Sistema mínimo microcontrolador PIC

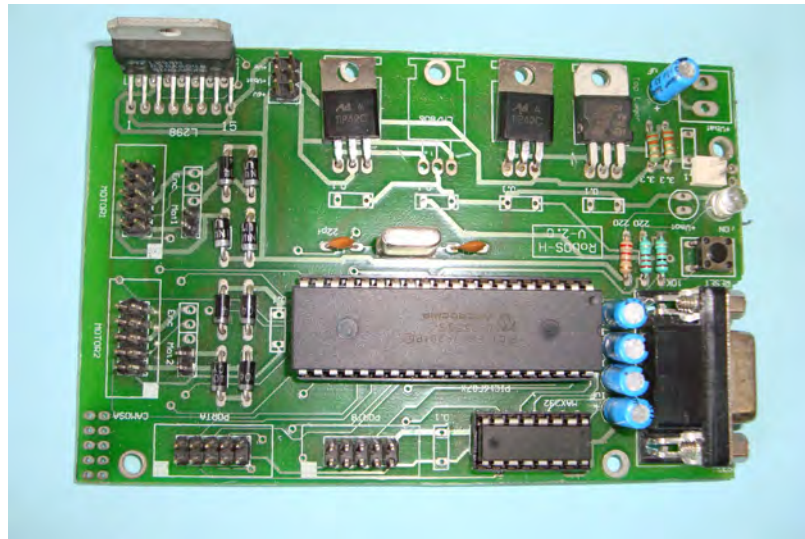


Figura 4.8: Tarjeta usada PIC16F877

4.5. Etapa de potencia

Son los circuitos empleados para proporcionar la corriente y la dirección que ésta circulará a través de los motores; en caso que la corriente requerida no sea muy alta, en aplicaciones de robótica se recomienda emplear circuitos integrados que sustituyen al alambrado de puentes H formado por transistores; reduciendo tanto el espacio como el costo de esta etapa, así como la complejidad para controlarla.

El circuito utilizado es el L298 que puede controlar el sentido de giro de dos motores de corriente directa además de contar con terminales específicas para ingresar señales que modulen el ancho de pulso (PWM), que permitirá variar la velocidad de giro.

La conexión con el microcontrolador se muestra en la figura 4.9. Para controlar un motor se requieren de dos señales de valor contrario (para generar la diferencia de potencial entre las terminales del motor) y la señal que permite que circule la corriente o denominada habilitador (comúnmente es conectada a la función PWM); de manera idéntica para el otro motor.

Se ha empleado el circuito L298; cuyas características más importantes son:

- Dos puentes H internos
- Tensión de alimentación de 5 volts
- Tensión independiente a motores de hasta 46 volts
- 4 amperes de salida
- Señales de control TTL
- Alta inmunidad al ruido

4.6. Sensores

Existen una gran variedad de sensores que se pueden agregar a un robot, dependiente de la función que este realizará entre los más usados se encuentran:

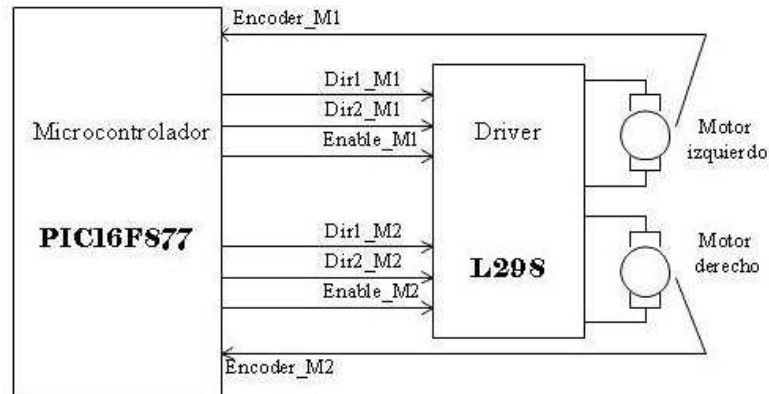


Figura 4.9: Control del circuito L298

- Sensores de contacto
- Sensores de luz
- Sensores ultrasónicos
- Sensores ópticos
- Cámaras
- Brújulas
- Laser

El objetivo principal del trabajo no contempla directamente el uso de sensores; no obstante se han adaptado sensores de contacto, infrarrojos y sonar para aplicaciones futuras. La arquitectura del robot tendrá la versatilidad de expandir su capacidad y conectar sensores extras.

4.6.1. Sensores de contacto

Son utilizados para saber cuando el robot ha tenido contacto con un objeto durante su recorrido; esta información servirá para generar acciones de control. El robot contiene un arreglo de 7 interruptores alrededor de su circunferencia.

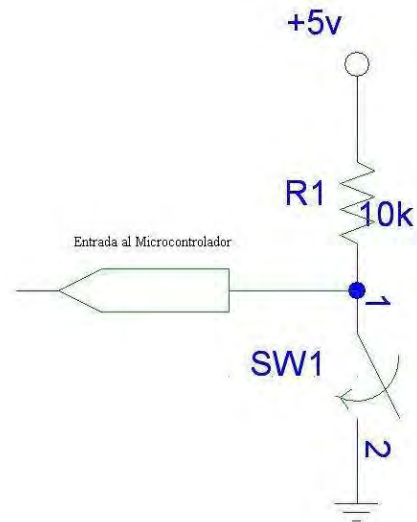


Figura 4.10: Electrónica microswitch

4.6.2. Infrarrojos

El objetivo de usar este tipo de sensores es la detección de objetos sin tener contacto con estos. En general está compuesto de un led infrarrojo que transmite un haz de luz infrarroja y un fotodetector (fototransistor o fotodiodo) que reciba la señal emitida por el transmisor, con la señal detectada se podrá actuar a tiempo para evitar coalisiones. El robot tiene tres sensores de este tipo.

Se han elegido por facilidad de uso el sensor GP2Y021YK, este proporciona una señal analógica dependiente de la distancia a la que se encuentra un objeto, el diagrama interno se presenta en la figura 4.11.

4.6.3. Sensor ultrasónico (sonar)

Con este tipo de sensores se evita el contacto directo con obstáculos, en ocasiones empleados para reconocimiento del entorno del robot. Se emite una señal sonora a cierta frecuencia, misma que al encontrar un objeto será rebotada; el tiempo que tarda en regresar esta señal servirá para calcular la distancia a la que se encuentra el obstáculo y actuar de acuerdo a los algoritmos programados.

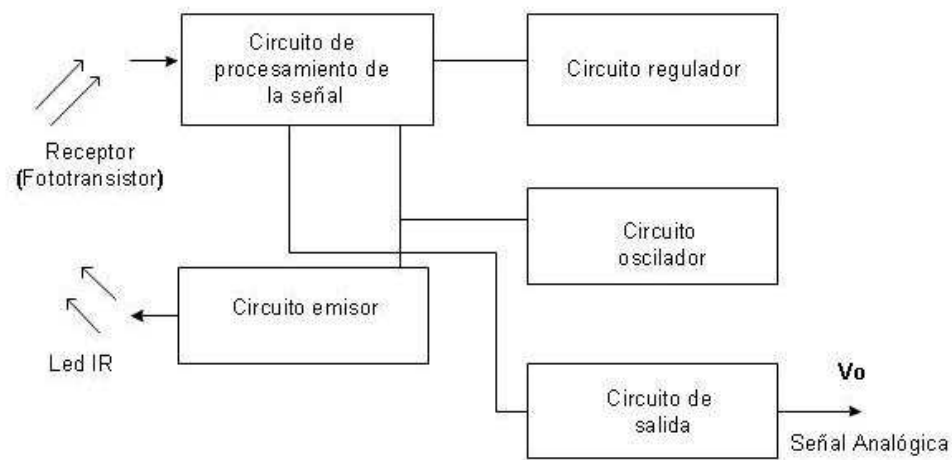


Figura 4.11: Diagrama a bloques sensor GP2Y0A21

Se han colocado dos tipos de sonares: los MaxSonar EZ1 y el módulo 6500 de polaroid, ambos procesan el eco obtenido por el retorno de la señal enviada proporcionando una señal analógica, producto de analizar el tiempo se envió la señal emisora y su recepción; en el caso de los primeros sensores la señal es enviada a 42 KHz y para los segundos a 49.4 KHz.

En general la distancia es calculada por la ecuación:

$$D = \frac{Tv}{2} \quad (4.5)$$

donde D es la distancia , T tiempo del eco y v la velocidad del sonido.

El MaxSonar EZ1 puede detectar objetos a una distancia de 6.45 metros. La salida es entregada como un ancho de pulso, como una señal analógica o una salida digital para transmitir a través de un formato serie asíncrono.

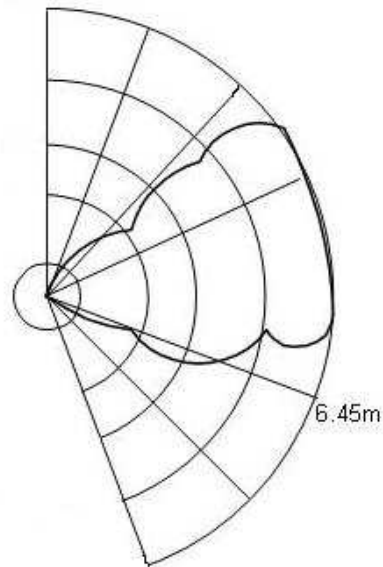


Figura 4.12: Distribución onda sonora

4.7. Software

La programación del robot que para la aplicación se ha incluido, son la generación del control de velocidad a través de la técnica de modulación de ancho de pulso (*PWM*); la función de captura de eventos externos para obtener la lectura de los encoders, el control de señales en tiempo real al utilizar técnicas de control en base a peticiones de interrupción de los temporizadores, el envío de comandos así como la transferencia de información desde o hacia el microcontrolador usando el protocolo serie asíncrono por medio de una terminal externa y por último la rutina del controlador *PID* que permitirá generar la velocidad deseada.

4.7.1. Modulación de ancho de pulso

Se genera una señal a un período constante y se modula el tiempo en alto de la misma; con lo que se obtendrá un ciclo de trabajo variable, en respuesta a esta acción el robot se podrá desplazar a diferentes velocidades.

Es usado el *TIMER2* y los módulos *CCP* del microcontrolador *PIC* para fijar la señal y controlar el *PWM*; el período está dado por (4.6):

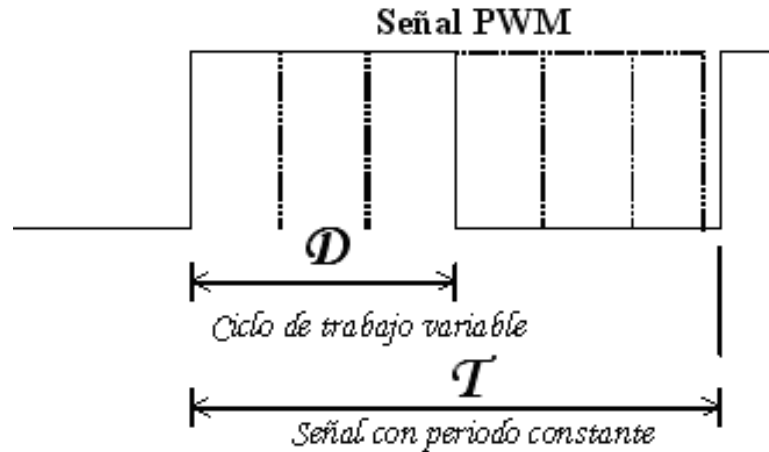


Figura 4.13: Modulación de ancho de pulso

$$T = (T_{osc_int})(PR2 + 1)(PREDIVISOR) \quad (4.6)$$

donde T_{osc_int} es el ciclo de máquina de procesador y es el inverso del reloj externo dividido por cuatro; $PR2$ valor cargado en un registro interno de ocho bits y el $PREDIVISOR$ es el valor seleccionado para que el registro TMR2 se incremente cada cantidad de ciclos de reloj configurados por este parámetro, pudiendo ser de 1, 4 o 16.

El oscilador externo es de 20 MHz, con lo cual se obtiene un ciclo de máquina de $0,2\mu s$; considerando un predivisor de uno y un valor en $PR2$ de 255, al usar la ecuación (4.6) se tendrá:

$$T = (0,2\mu s)(PR2 + 1)(1) = 51,2\mu s$$

por lo tanto la frecuencia de la señal será:

$$f = 19,531KHz$$

Para ser programado usando el compilador CCS se usa la función:

$$SETUP_TIMER_2(T2_DIV_BY_1, 255, 10);$$

$T2_DIV_BY_1$ indica el predivisor de uno, 255 el valor que tendrá PR2 y el número 10 es usado para generar peticiones de interrupción.

El tiempo de la señal en alto, es calculado con la ecuación 4.7 cuando sea empleado una resolución de 8 bits.

$$T_{ON} = (VALOR) \left(\frac{1}{XTAL} \right) (4) (PREDIVISOR) \quad (4.7)$$

Para una resolución de 10 bits se usará 4.8.

$$T_{ON} = (VALOR) \left(\frac{1}{XTAL} \right) (PREDIVISOR) \quad (4.8)$$

4.7.2. Captura de pulsos (encoders)

La lectura de los pulsos se realiza con las funciones del TIMER0 y TIMER1; estos temporizadores pueden ser configurados como temporizadores o como acumuladores de eventos externos; en esta última modalidad, la información que generan los encoders permitirán conocer la velocidad de rotación de los motores. Con el valor de los encoders y el radio de la llanta servirán para estimar la distancia recorrida por el robot y la velocidad angular de las llantas.

En ambos temporizadores se requiere configurar el tipo de flanco que incrementará los registros internos asociados a estos, la figura 4.14 muestra la adecuación de los encoders al microcontrolador.

La función de configuración empleada para la detección de flancos de subida en el motor izquierdo es:

$$SETUP_TIMER_1(T1_EXTERNAL|T1_DIV_1);$$

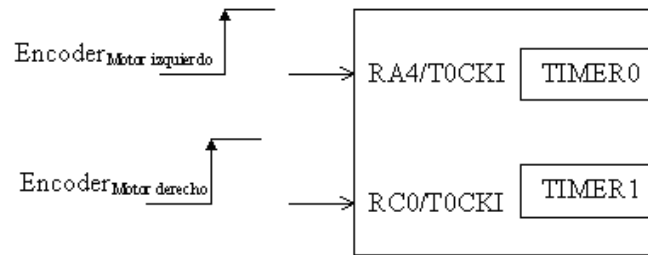


Figura 4.14: Captura de pulsos

Para la configuración de la detección del mismo tipo de pulsos en el motor derecho se emplea:

```
SETUP_TIMER_0(RTCC_EXTERNAL_L_TO_H|T1_DIV_1);
```

La lectura del encoder izquierdo será adquirida con:

```
CntLeft = get_timer0();
```

y para el derecho:

```
CntRight = get_timer1();
```

4.7.3. Interrupción (TIMER2)

El período de interrupción es controlado por el TIMER2, este será usado para generar el tiempo de muestreo para la actualización de la lectura de los encoders; además de las configuraciones descritas en el PWM se utiliza un POSTESCALADOR, el cuál indicará el periodo de la petición de interrupción del TIMER2; la expresión quedará ahora como:

$$T = (T_{osc_{int}})(PR2)(PREDIVISOR)(POSTESCALADOR) \quad (4.9)$$

Cuando el valor del registro TMR2 sea igual a PR2 entonces se genera una comparación exitosa; cuando se han cumplido la cantidad de comparaciones indicadas en el POSTESCALADOR, entonces se genera una solicitud de interrupción. Para configurar el periodo de interrupción se usa la función:

SETUP_TIMER_2(T2_DIV_BY_1, 255, 10);

Especifica que se usará un predivisor de uno, el valor de PR2 es 255 y el postescalador de 10; con lo que se obtiene un tiempo de $512\mu s$. La actualización de la lectura de los encoders se realiza cada 97.7 ms, por lo que son requeridas 192 interrupciones para cubrir este período.

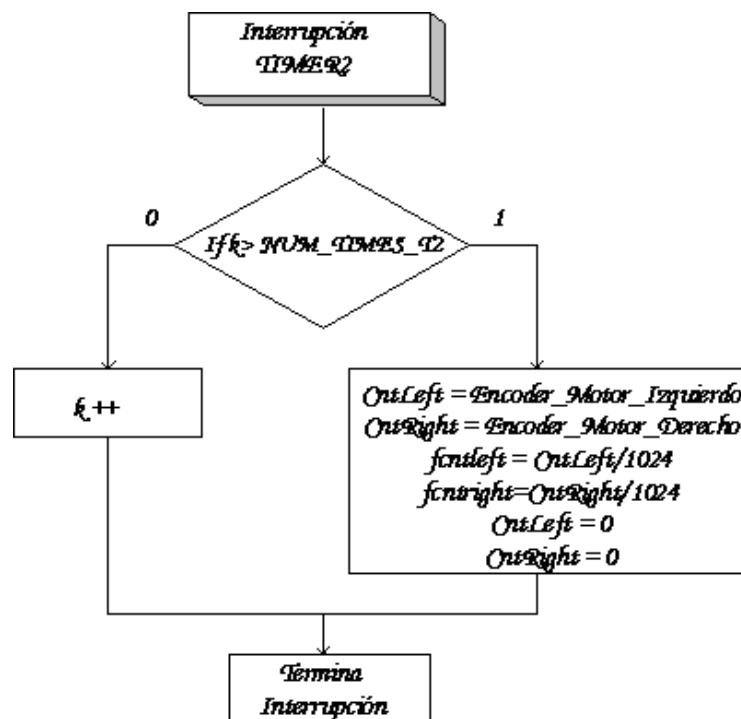


Figura 4.15: Rutina de interrupción TIMER2

4.7.4. Velocidad de entrada

Se usan las interrupciones por recepción de datos del puerto serie para indicar los comandos de dirección y velocidad del robot; como los datos son recibidos en formato ASCII se han programado funciones que permiten cambiar este formato a números

decimales para manejo interno del programa.

La figura 4.16 describe el procedimiento.

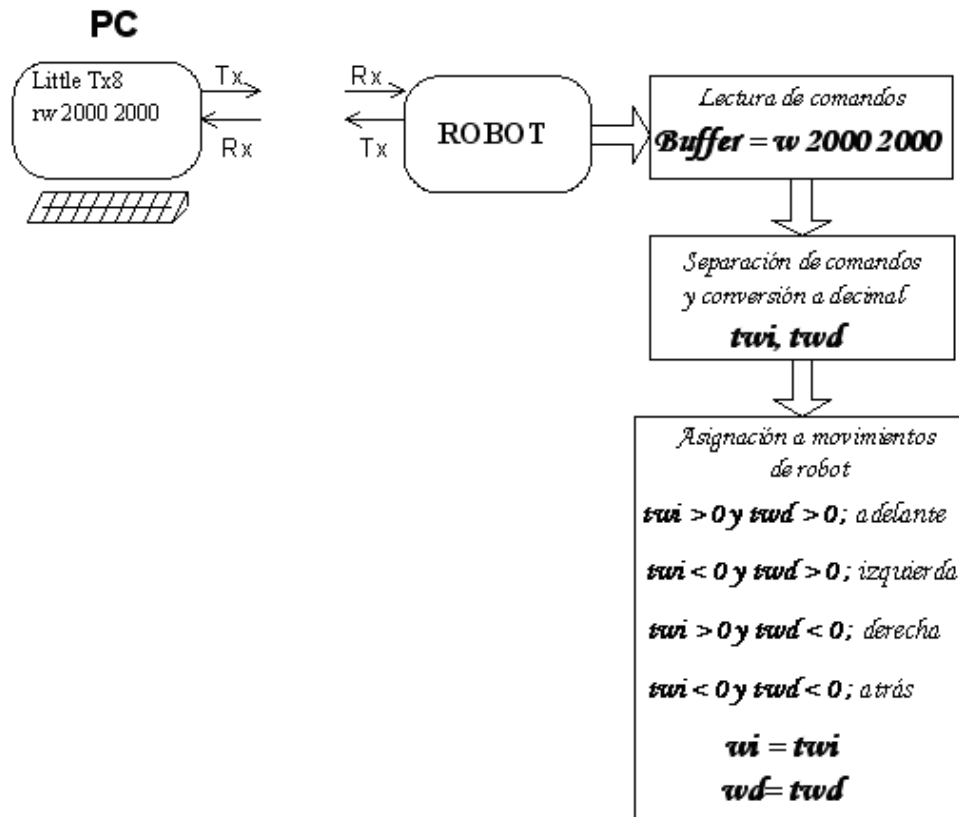


Figura 4.16: Recepción de datos

4.7.5. Algoritmo recursivo PID

Al ser ingresados los comandos de movimiento al robot, se ejecuta el controlador PID interno, con el objeto de alcanzar y mantener la velocidad deseada; este controlador es representado por la figura 4.17.

El diagrama de flujo que describe al controlador es el que se muestra en la figura 4.18:

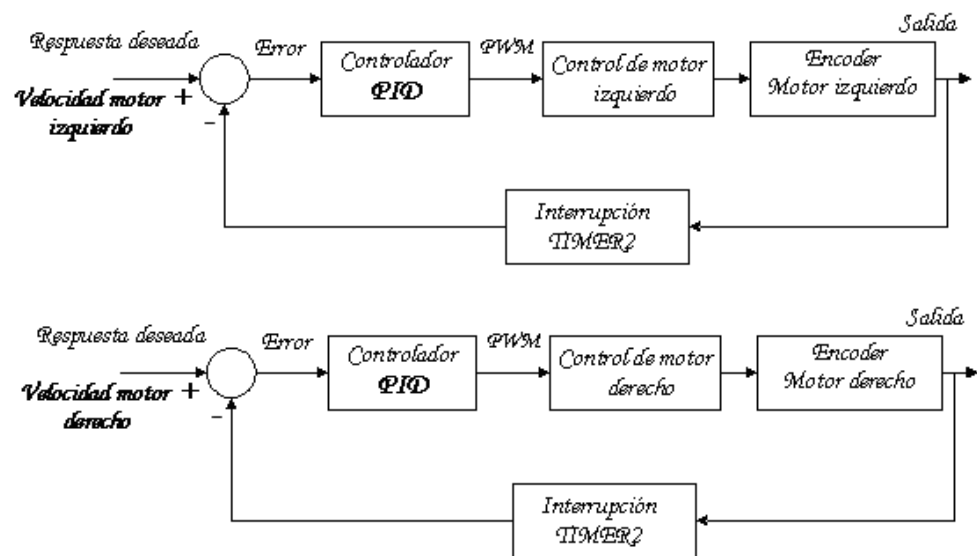


Figura 4.17: Controlador PID

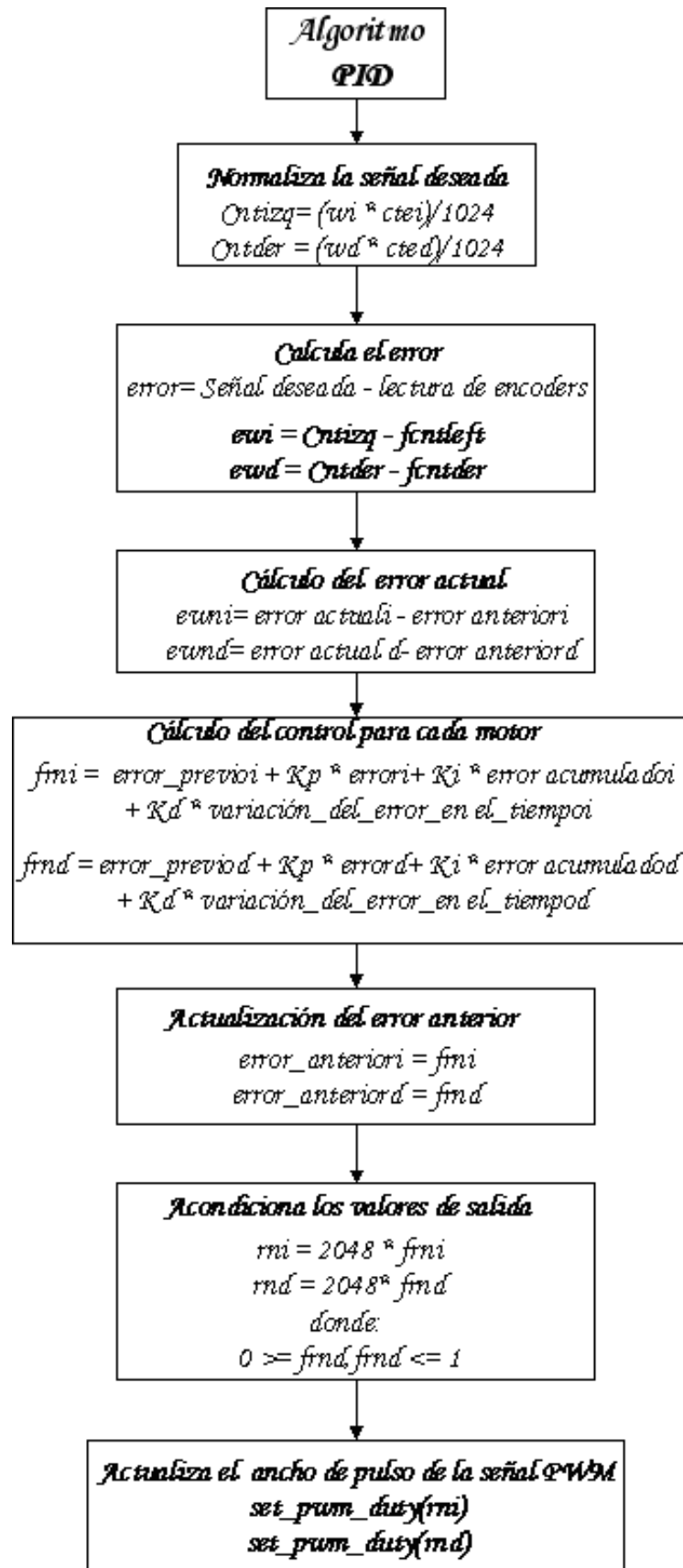


Figura 4.18: Algoritmo PID

Capítulo 5

Estructura del filtro

Como se ha planteado en los capítulos anteriores, el objetivo es calcular la función de transferencia del robot móvil construido para tal efecto, de las estructuras descritas en el capítulo 2, fué utilizada la modalidad de identificación de sistemas en la que permita de manera experimental realizar su estimación con un alto grado de aceptación.

5.1. Análisis de la estructura del modelo

La función de transferencia discreta encontrada en el capítulo 3 (ecuaciones 3.28 y 3.29), permite la propuesta del orden y el tipo de filtro usado para modelar (FIR o IIR), aunque se aproxima a la apariencia de un filtro de respuesta finita.

5.2. Estimación de los parámetros

La estructura del filtrado adaptable en la modalidad de identificación de sistemas descrito previamente, se presenta en la figura 5.1. En caso de emplear un filtro de tipo FIR, tan solo podrán existir los coeficientes $b(k)s$; por otro lado, al modelar con un filtro IIR la función de transferencia será integrada por los coeficientes $b(k)s$ y los coeficientes $a(k)s$.

Mediante el algoritmo adaptable se ajustarán y actualizarán los coeficientes del filtro para igualar a los coeficientes del sistema desconocido (robot móvil) y cuando se obtenga se logre esto, se pueda considerar que se ha logrado identificar el sistema y como consecuencia se haya encontrado la función de transferencia.

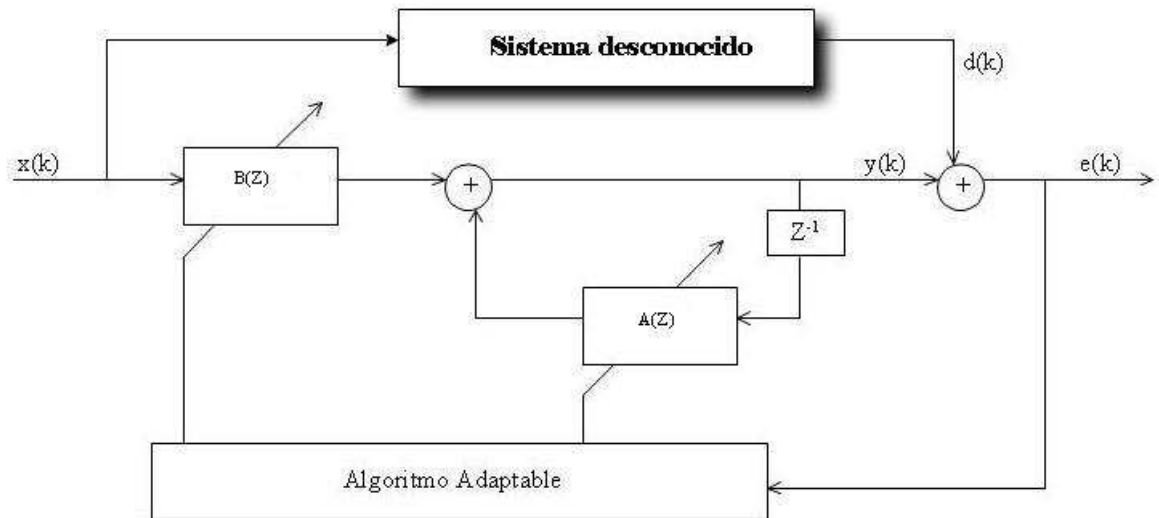


Figura 5.1: Identificación de sistemas

5.2.1. Estimación usando filtro FIR

El algoritmo empleado fué el algoritmo LMS, el cuál se describe en el capítulo 2: ecuaciones (2.54), (2.55), (2.56) y se representa en la figura 5.2.

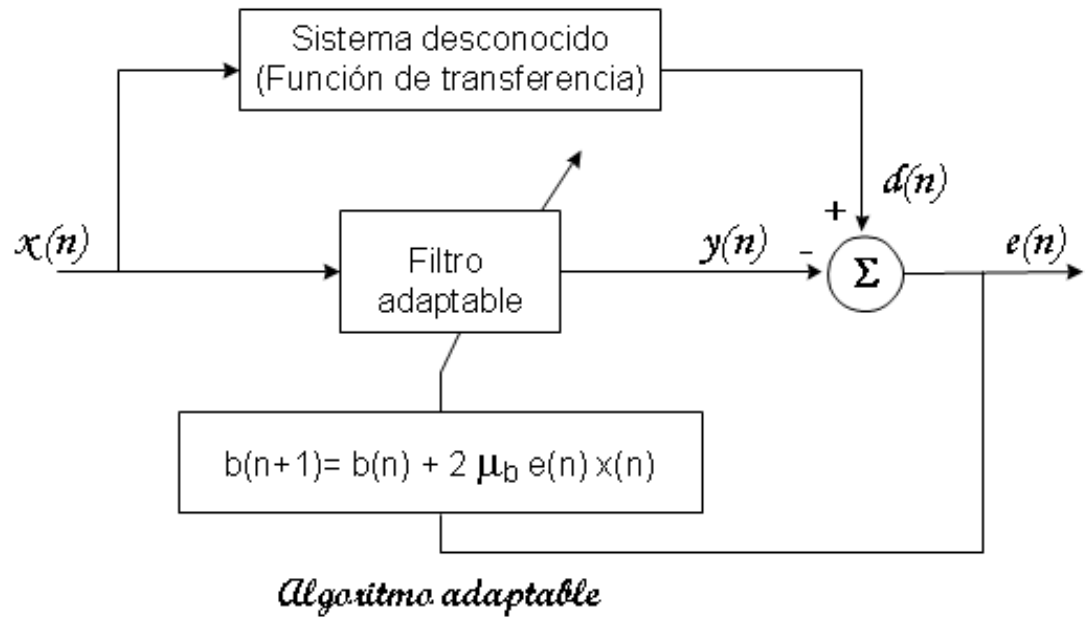


Figura 5.2: Algoritmo LMS Filtro FIR

5.2.2. Estimación usando filtro IIR

La identificación empleando un filtro IIR hace presente ambos grupos de coeficientes ($a(k)$'s y $b(k)$'s).

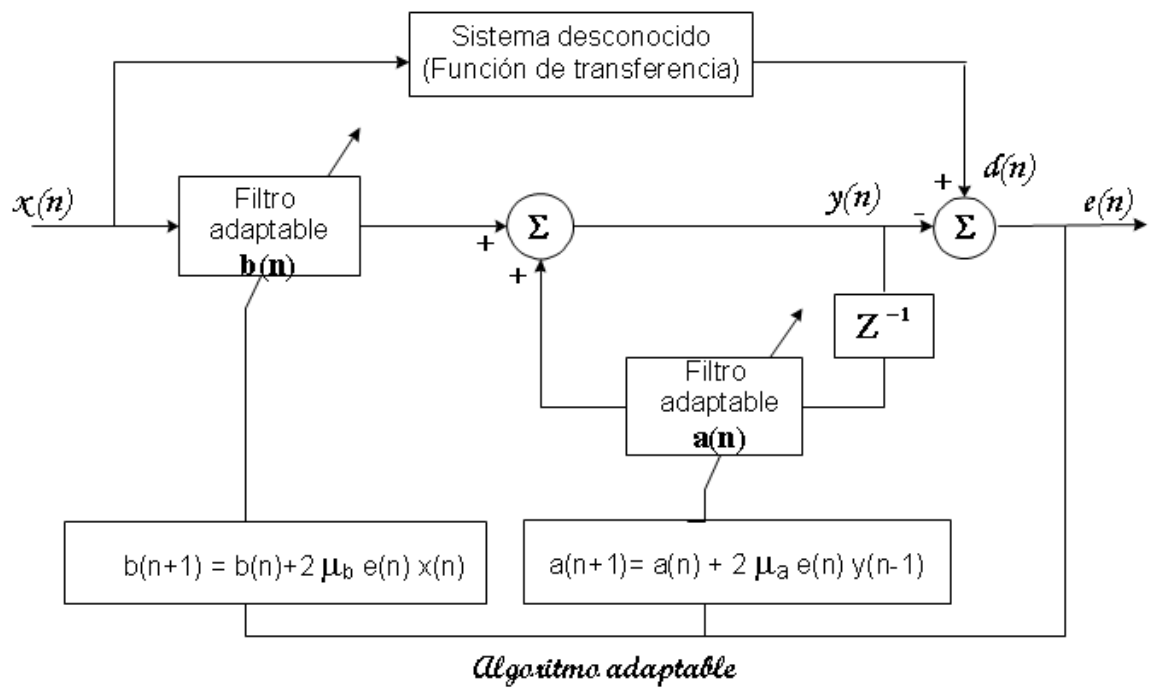


Figura 5.3: Algoritmo LMS Filtro IIR

Capítulo 6

Pruebas y análisis

La identificación o modelado de sistemas consiste en adecuar las siguientes etapas:

- Planteamiento experimental
- Propuesta de la estructura del modelo
- Cálculo de los parámetros
- Validación del modelo

6.1. Realización de experimentos

Utilizando las funciones descritas y programadas en el capítulo 4, se han realizado múltiples pruebas donde se analiza el comportamiento del robot a señales de control con velocidades fijas, usando diferentes ciclos de trabajo, controlando con la técnica de modulación de ancho de pulso.

También fueron realizadas pruebas en las que se incrementaba la velocidad en intervalos del 10 %, iniciando con velocidad cero hasta llegar al máximo, para proceder al decremento hasta llegar nuevamente a una velocidad nula; esta prueba fué utilizada para el análisis de datos.

El robot fué puesto sobre el piso, con la finalidad de capturar la cantidad de pulsos recorridos por cada motor, con lo que se conocerá la velocidad angular de estos y con el radio de las llantas se calculará la velocidad angular real a la que están girando.

La lectura de los encoders es transmitida a través del protocolo serie asíncrono; en eventos preliminares a una computadora personal usando un cable para la recepción de datos, siendo esto un tanto ineficiente e incómodo debido a que el robot debe operar sobre el piso y es necesario capturar la información completa de su comportamiento.

Para evitar este problema se ha montado sobre el robot una pocket pc, con la aplicación que permite recibir datos a través del protocolo serie y guardar la información en la memoria de este dispositivo; con lo que se obtienen datos más confiables y directos del robot (figura 6.1).



Figura 6.1: Prototipo de prueba

Una vez obtenida la lectura de los encoders, se procesa para calcular la velocidad angular de las llantas para tener el formato de datos requeridos para el análisis en matlab.

El comportamiento obtenido se aprecia en las figuras 6.2 y 6.3.

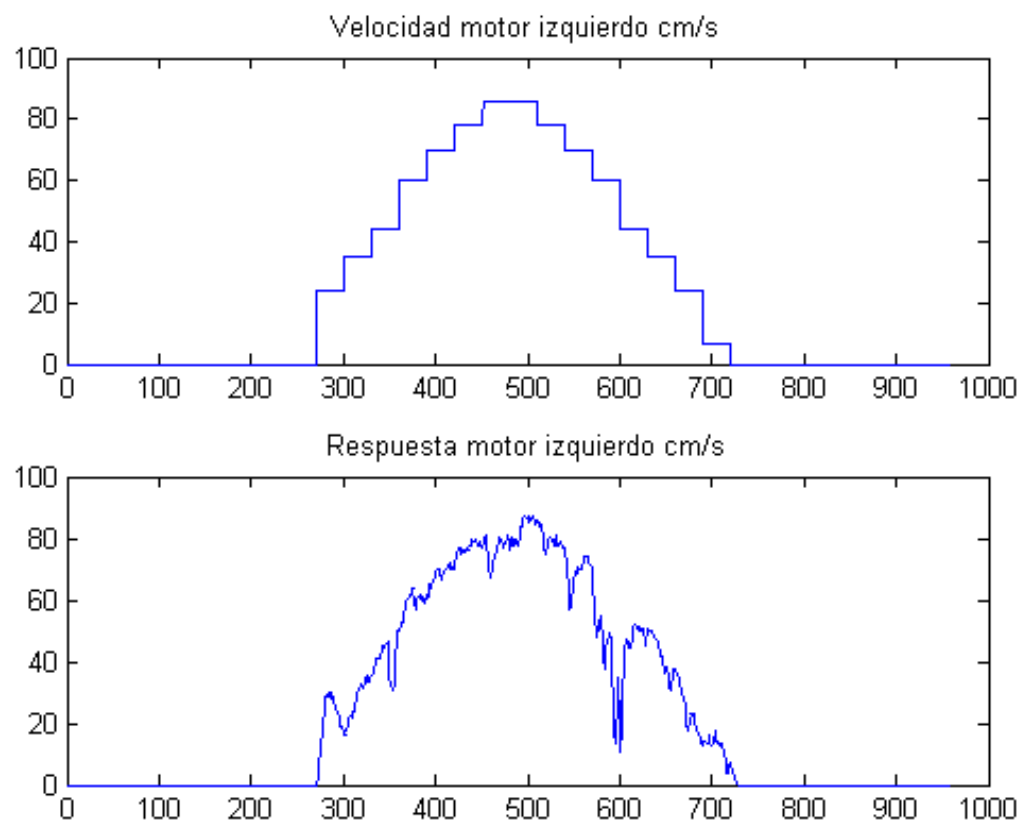


Figura 6.2: Velocidad de entrada deseada motor izquierdo

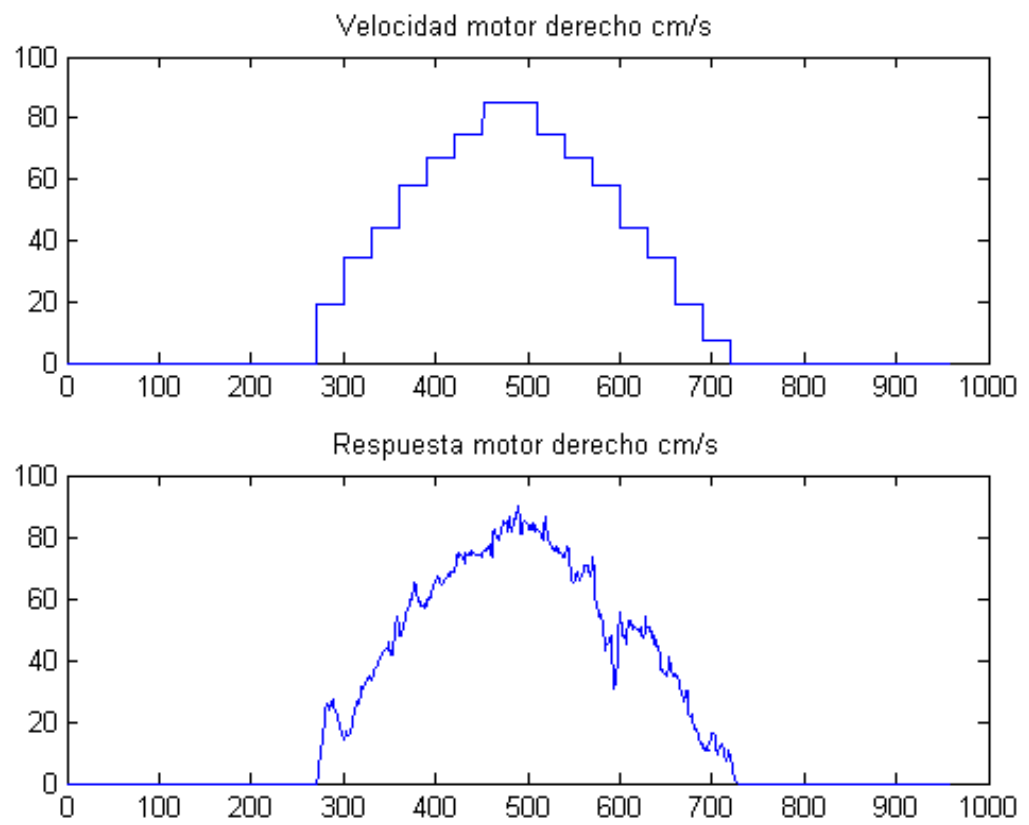


Figura 6.3: Velocidad de entrada deseada motor derecho

6.2. Resultados

La función de transferencia discreta encontrada en el capítulo 3 (ecuaciones 3.28 y 3.29), permite la propuesta del orden y el tipo de filtro usado para modelar (FIR o IIR). Es de hacer notar que se parece a un filtro FIR que en este caso es posible realizar debido a que se dispone de un conjunto amplio de muestras.

En caso que las muestras adquiridas sean reducidas; es conveniente la estimación usando la estructura de un filtro IIR.

Se han programado los scrips en matlab para calcular el valor de los coeficientes en los cuales se implataron los algoritmos adaptables para los filtros FIR e IIR(se anexan en el apéndice).

6.2.1. Respuesta estimando con un filtro FIR

El algoritmo empleado fué el algoritmo LMS; el cuál se describe en el capítulo 2 (ecuaciones (2.54), (2.55), (2.56) y se representa en la figura 5.2.

La ejecución del algoritmo para el modelado con un filtro FIR con cuatro coeficientes se muestra en las figuras 6.4 y 6.5 para cada una de las llantas. En dichas gráficas se podrá notar la actualización que van adquiriendo los coeficientes $b(k)$ s, así como el error del sistema el cual se va reduciendo; permitiendo la convergencia entre los coeficientes del filtro adaptable y del modelo del robot.

El valor de los coeficientes obtenidos en el cálculo de la velocidad angular de la llanta izquierda se muestra en la tabla 6.1, la función de transferencia por la ecuación (6.1) y su respuesta por la ecuación 6.2.

Coefficiente	wb0	wb1	wb2	wb3
<i>Ceros</i>	0.25278	0.24958	0.24714	0.24597

Tabla 6.1: Coeficientes filtro FIR llanta izquierda

$$H(Z) = 0.25278 + 0.24958Z^{-1} + 0.4714Z^{-2} + 0.24597Z^{-3} \quad (6.1)$$

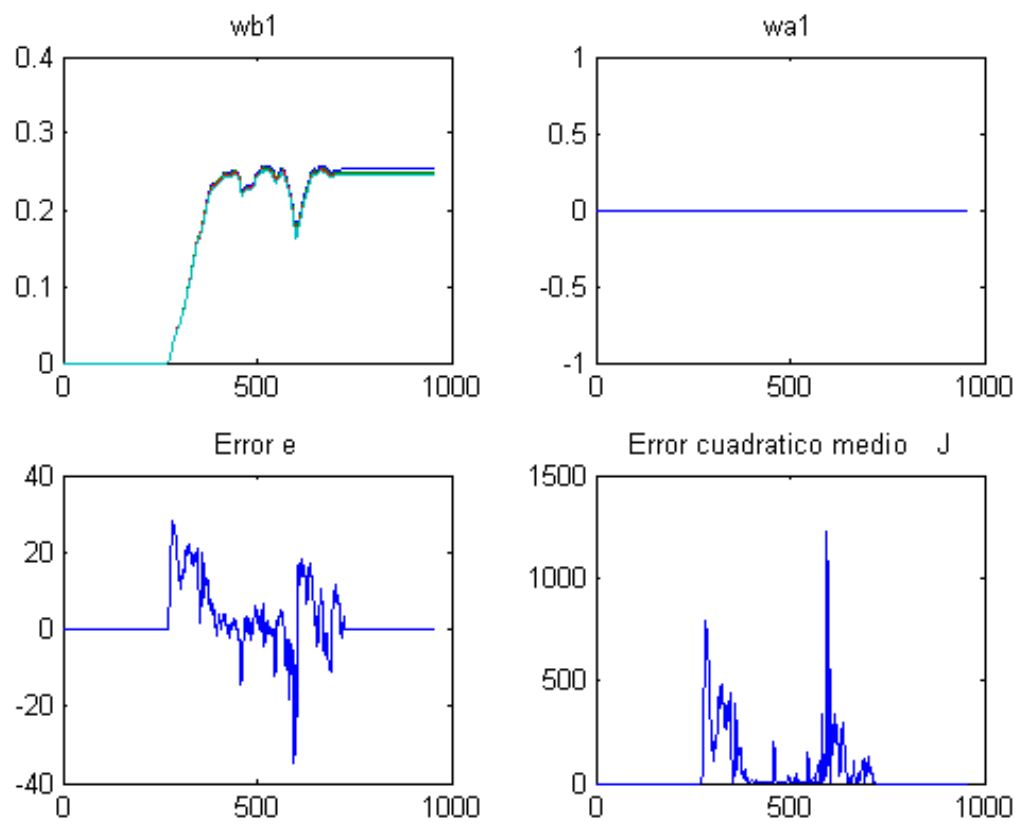


Figura 6.4: Comportamiento del filtro adaptable motor izquierdo (FIR)

$$y[n] = 0.25278x[n] + 0.24958x[n - 1] + 0.4714x[n - 2] + 0.24597x[n - 3] \quad (6.2)$$

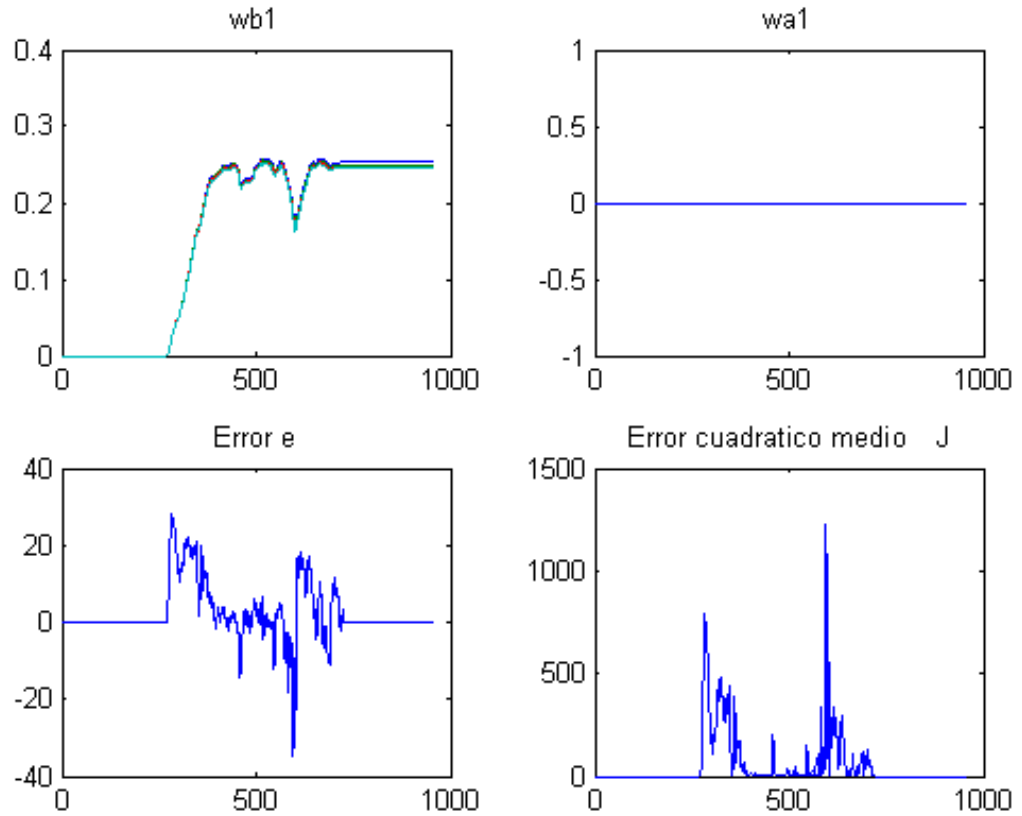


Figura 6.5: Comportamiento del filtro adaptable motor derecho (FIR)

Los coeficientes para el lado derecho, la función de transferencia y la salida del sistema son los descritos en la tabla 6.2 y las ecuaciones 6.3 y 6.4 respectivamente.

$$H(Z) = 0.27311 + 0.27215Z^{-1} + 0.27162Z^{-2} + 0.27141Z^{-3} \quad (6.3)$$

$$y[n] = 0.27311x[n] + 0.27215x[n - 1] + 0.27162x[n - 2] + 0.27141x[n - 3] \quad (6.4)$$

Coefficiente	wb0	wb1	wb2	wb3
<i>Ceros</i>	0.27311	0.27215	0.27162	0.27141

Tabla 6.2: Coeficientes filtro FIR llanta derecha

6.2.2. Respuesta estimando con un filtro IIR

La identificación empleando un filtro IIR hace presente ambos grupos de coeficientes ($a(k)$'s y $b(k)$'s) tal como se muestra en la figura 5.3 y la evolución de ellos en las figuras 6.6 y 6.7.

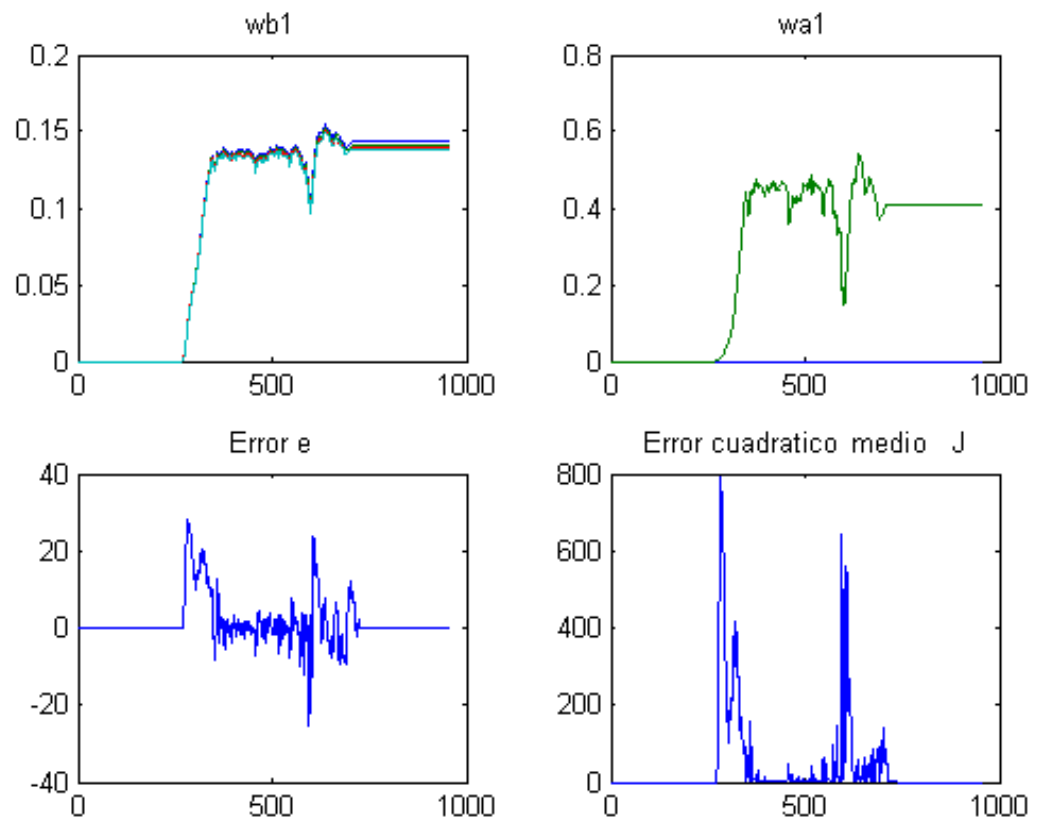


Figura 6.6: Comportamiento del filtro adaptable motor izquierdo (IIR)

Los coeficientes encontrados se muestran en la tabla 6.3, la función respectiva en

la ecuación (6.5) y la salida por (6.6).

Coefficiente	0	1	2	3
Wb	0.14319	0.14076	0.13901	0.13853
Wa	0	0.40959	-	-

Tabla 6.3: Coeficientes filtro IIR llanta izquierda

$$H(Z) = \frac{0.14319 + 0.14076Z^{-1} + 0.13901Z^{-2} + 0.13853Z^{-3}}{1 + 0.40959Z^{-2}} \quad (6.5)$$

$$y[n] = 0.14319x[n] + 0.14076x[n-1] + 0.13901x[n-2] + 0.13853x[n-3] - 0.40959y[n-2] \quad (6.6)$$

Obteniendo el conjunto de parámetros de la tabla 6.4; la función de transferencia se representa en la ecuación (6.7) y la salida discreta por la ecuación (6.8)

Coefficiente	0	1	2	3
Wb	0.14123	0.14105	0.14098	0.14096
Wa	0	0.49004	-	-

Tabla 6.4: Coeficientes filtro IIR llanta derecha

$$H(Z) = \frac{0.14123 + 0.14105Z^{-1} + 0.14098Z^{-2} + 0.14096Z^{-3}}{1 + 0.49004Z^{-2}} \quad (6.7)$$

$$y[n] = 0.14123x[n] + 0.14105x[n-1] + 0.14098x[n-2] + 0.14096x[n-3] - 0.49004y[n-2] \quad (6.8)$$

6.3. Validación

La validación consiste en la comprobación de las funciones de transferencia encontradas excitandolas con la entrada usada en la identificación.

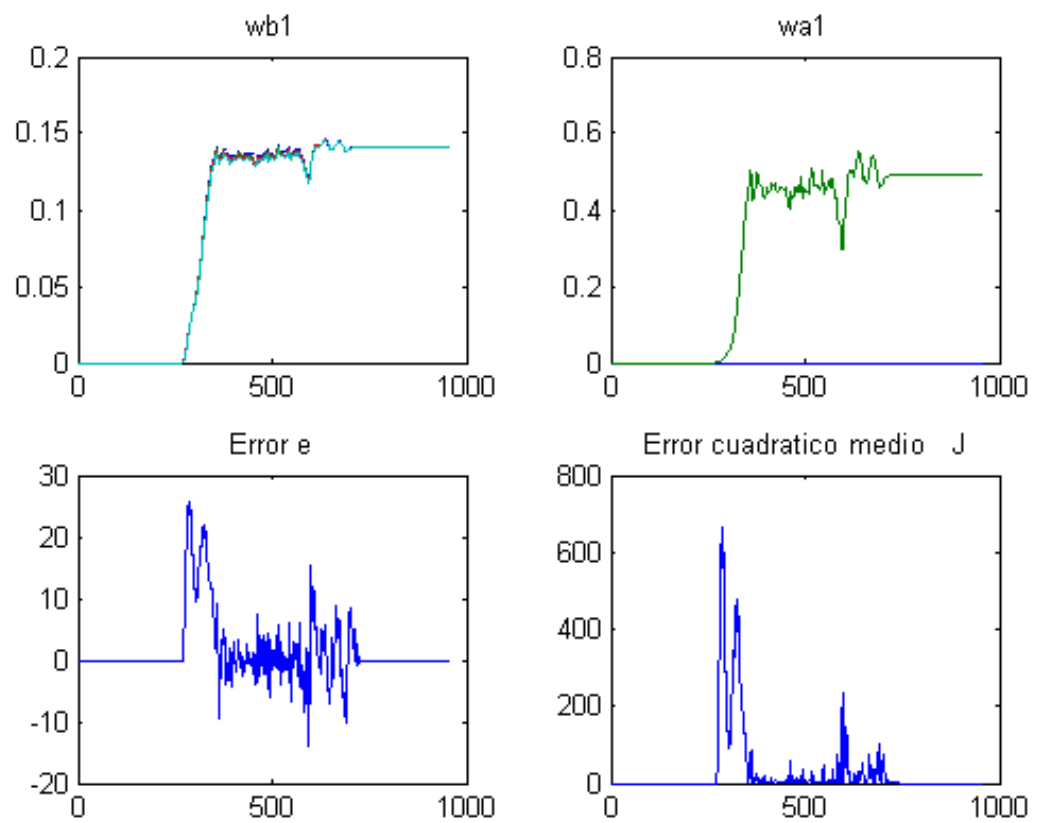


Figura 6.7: Comportamiento del filtro adaptable motor derecho (IIR)

6.3.1. Validación filtro FIR

La respuesta del comportamiento de la velocidad angular de las llantas usando los coeficientes encontrados con el filtro tipo FIR se muestra en la figura 6.8 para la izquierda y la figura 6.9 para la derecha.

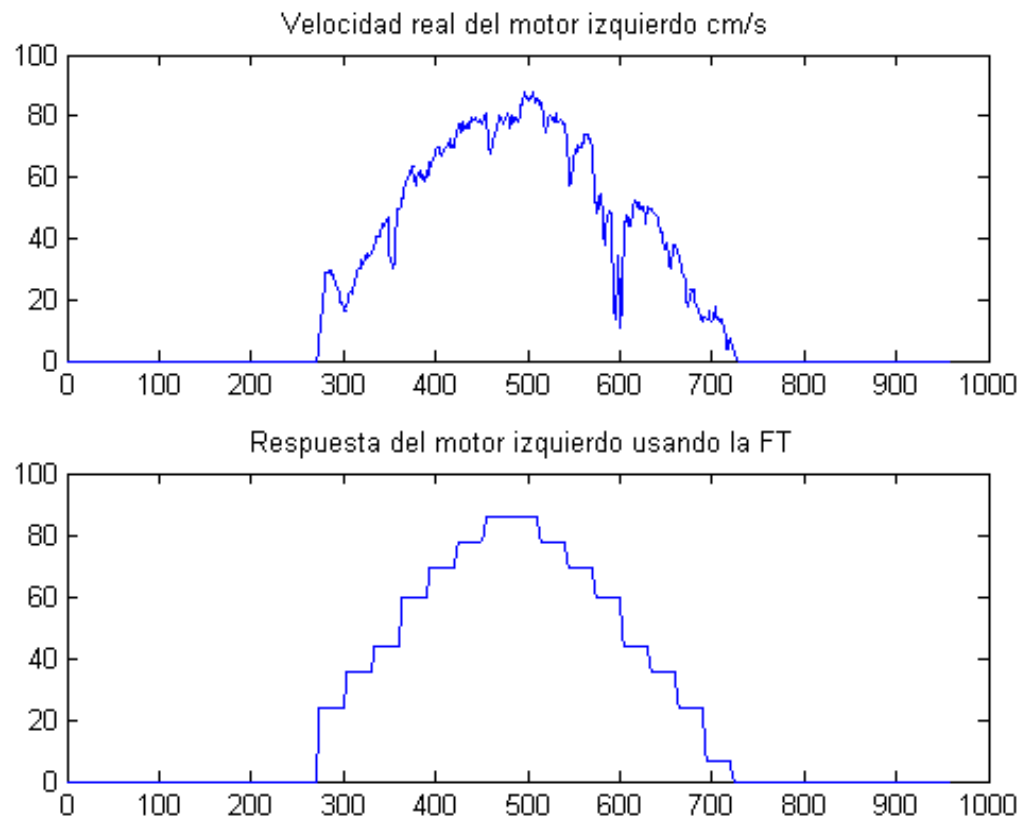


Figura 6.8: Respuesta de la función encontrada para el motor izquierdo (FIR)

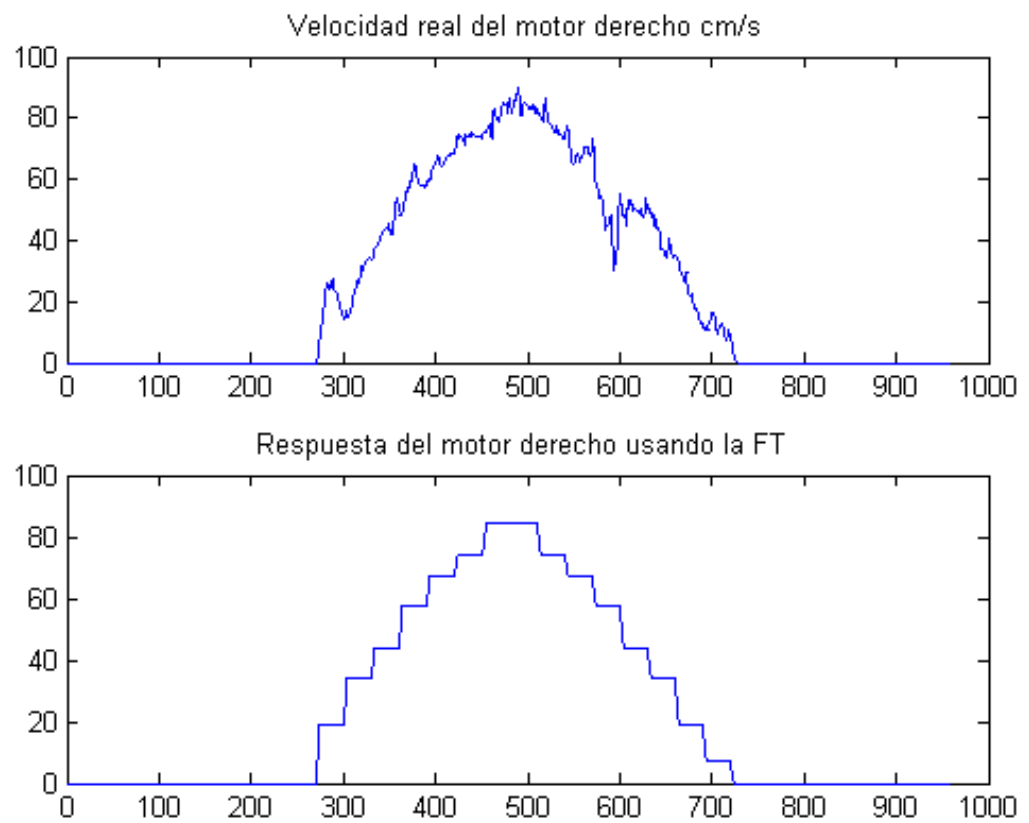


Figura 6.9: Respuesta de la función encontrada para el motor derecho (FIR)

6.3.2. Validación filtro IIR

Utilizando la función de transferencia obtenida con el filtro IIR con el conjunto de datos sugeridos como entrada genera la gráfica inferior de las figuras 6.10 y 6.11.

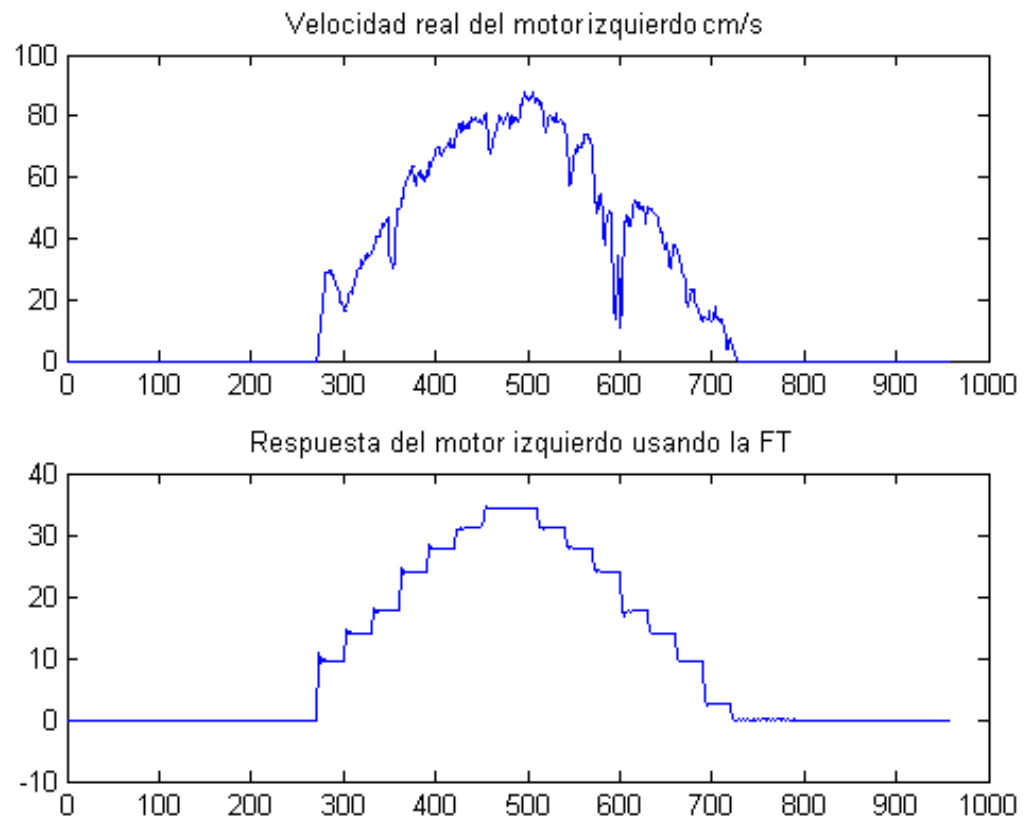


Figura 6.10: Respuesta de la función encontrada para el motor izquierdo (IIR)

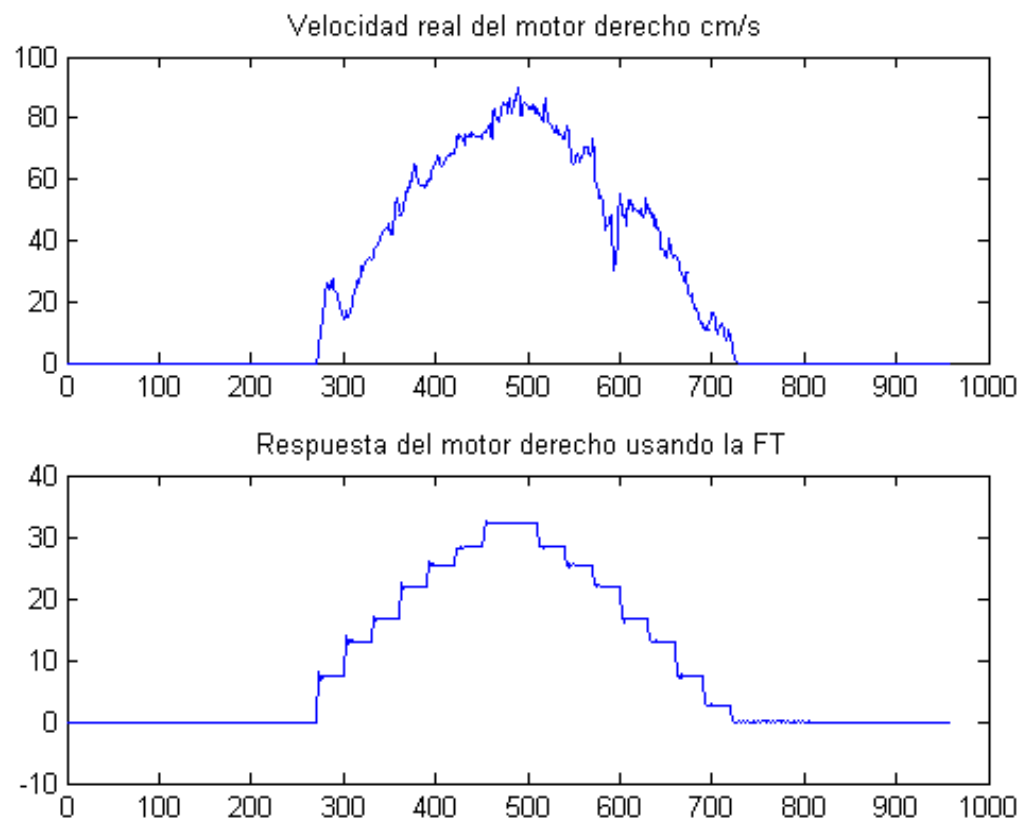


Figura 6.11: Respuesta de la función encontrada para el motor derecho (IIR)

Por lo tanto, la función de transferencia que produce el comportamiento esperado se obtuvo por medio del filtro FIR; las figuras 6.12 y 6.13 presentan las gráficas más representativas para cada caso.

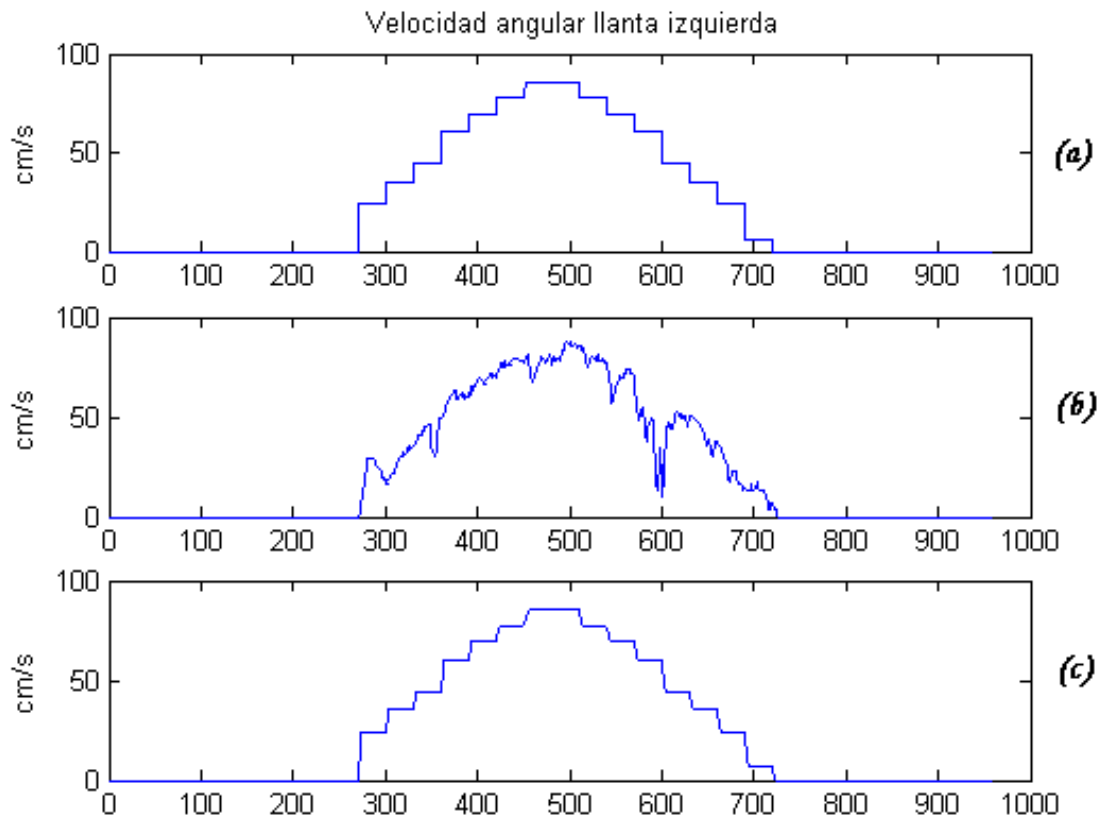


Figura 6.12: Velocidad angular llanta izquierda

donde (a) es la entrada al sistema (respuesta deseada), (b) la respuesta del robot y (c) la respuesta del filtro utilizando la función obtenida para la llanta izquierda.

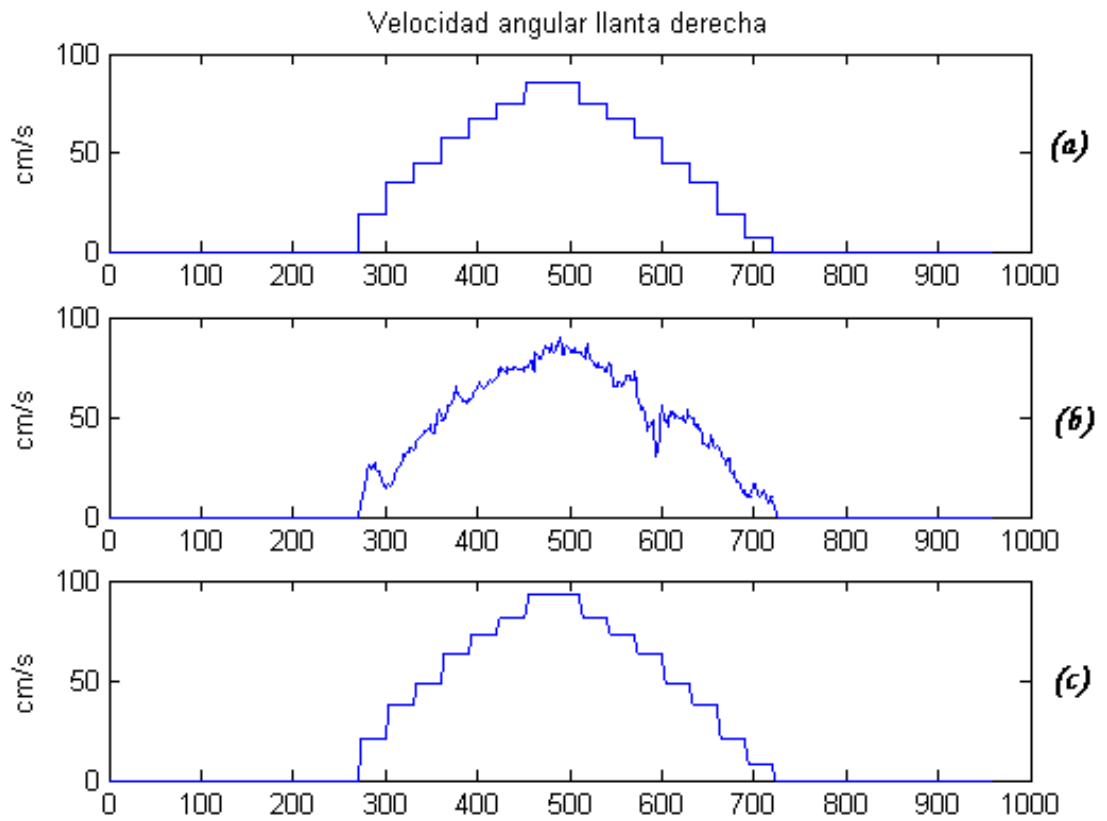


Figura 6.13: Velocidad angular llanta derecha

donde (a) es la entrada al sistema (respuesta deseada), (b) la respuesta del robot y (c) la respuesta del filtro utilizando la función obtenida para la llanta derecha.

Capítulo 7

Conclusiones

Utilizar la teoría de filtrado adaptable para realizar este trabajo fué bastante interesante ya que bajo un concepto relativamente sencillo y a la vez poderoso; se pueden realizar aplicaciones de una complejidad muy alta.

Se comprobó que efectivamente se puede emplear la identificación de sistema para resolver el objetivo planteado.

Como fué recalcado, debido a la estructura final del robot, involucra una cantidad grande de elementos; que hace difícil calcular analíticamente todos los parámetros físicos que lo integran y como consecuencia también la función de transferencia total; ya sea para controlar la velocidad lineal o la velocidad angular del robot.

Se obtuvo una función equivalente, que reproduce en gran medida el comportamiento del sistema.

Una vez obtenida la función de transferencia, es posible programar los controladores aplicables según sea el caso; reduciendo o eliminando errores y se obtenga un control más fiel. Un reto que surge de este trabajo es hacer que el robot actualice de manera dinámica los coeficientes para que sea capaz de adaptarse a su entorno ya sea en sus comandos de movimiento o en la lectura de los sensores.

Los microcontroladores son dispositivos que aportan la mayoría de los periféricos requeridos para controlar los robots móviles reduciendo substancialmente el tamaño del

diseño final; el microcontrolador PIC16F877 fué suficiente en este trabajo. En aplicaciones futuras que este procesador se vea excedido, es posible intercambiar directamente a una versión con más recursos.

Referencias

- [1] Bernard Widrow, Samuel D. Stearns. *Adaptive Signal Processing*. Prentice Hall Signal Processing Series.
- [2] Simon Haykin. *Adaptive Filter Theory*. Prentice Hall, Third Edition.
- [3] John R. Treichler, C. Richard Johnson, Jr, Michel Larimore. *Theory and Design of Adaptive Filters*. A Wiley-Interscience Publication.
- [4] Lennart, Ljung. *System Identification Theory for the user*. Prentice Hall, Information and System Sciences Series.
- [5] Emilio Sosa, Marcelino Tovar, Vicente Francés, Gustavo Camps. *Tratamiento Digital de Señales*. Prentice Hall.
- [6] Simon Haykin. *Modern Filter*. Macmillar Publishing Company.
- [7] Benjamin C. Kou. *Automatic Control System*. Prentice Hall, seventh edition.
- [8] Katsuhito Ogata. *Modern Control System*. Prentice Hall, third edition.
- [9] Microchip. *PIC16F877 Data Sheet* . www.microchip.com.
- [10] Jesús Savage Carmona. *Notas curso Robots móviles y Agentes Inteligentes* .Universidad Nacional Autónoma de México.
- [11] Jesús Savage Carmona. *Notas curso Construcción de Robots Móviles* . Universidad Nacional Autónoma de México.
- [12] Jesús Savage Carmona, Rubén Anaya. *"Prácticas Construcción de Robots Móviles* . Universidad Nacional Autónoma de México.
- [13] Rubén Anaya García. *Notas curso Microcomputadoras* . Universidad Nacional Autónoma de México.
- [14] Jesús Savage Carmona, Alberto Templos, Rubén Anaya. *Prácticas Laboratorio de Microcomputadoras* .Universidad Nacional Autónoma de México.