

Modelado y Control de un Propulsor para Robot Submarino Tele-operado (ROV)

Delgado Ramírez Guillermo, Ronquillo Lomeli Guillermo, Salgado Jiménez Tomás
memo_taxco@hotmail.com, gronquillo@cidesi.mx, tsalgado@cidesi.mx.

Centro de Ingeniería y Desarrollo Industrial (CIDESI).

Resumen

Debido al interés en la exploración de nuestros mares y ríos se han ido generando nuevas herramientas que ayudan a realizar dichas tareas. Una de las más relevantes son los ROVs; los cuales, provistos de diversos sensores, manipuladores y propulsores, logran llegar a lugar a los que antes simplemente era imposible. Referidos a estos últimos (los propulsores) son de vital importancia dado que ellos son los que proveen de movimiento, dirección y orientación al robot.

En el siguiente artículo se presenta un método para identificar y controlar los propulsores antes mencionados. La metodología está basada en modelado del propulsor a partir del comportamiento que éste tiene ante la inducción de un voltaje dado. La velocidad angular es almacenada y posteriormente procesada en un algoritmo de mínimos cuadrados, dando como resultado una función de transferencia; ésta sirve para programar y simular un control de velocidad PID continuo que sirve para tener una panorámica del comportamiento del propulsor al aplicarle un control. Posteriormente se aplica al propulsor un control discreto en tiempo real diseñado y sintonizado en LabVIEW.

Palabras clave: Propulsores de ROV's, método de mínimos cuadrados, identificación y control.

1. Introducción

Desde hace mucho tiempo ha existido el interés y la necesidad del hombre por explorar las profundidades de los mares, océanos, lagos y ríos. Para lograr esto se hace necesaria la investigación en el desarrollo de equipos como los robots submarinos, los cuales constituyen una herramienta muy útil y poderosa que facilita el estudio de variables marinas. Una variante importante de los robots submarinos son

los ROV (Remotely Opered Vehicles, Vehículos Operados Remotamente) [2, 7]. Los cuales son robots submarinos que están ligados a la superficie con un cable y son controlados remotamente. En CIDESI, se está trabajando en un proyecto enfocado al diseño y fabricación de un prototipo de ROV. Este proyecto es patrocinado por CONACYT y CIDESI). El nombre del robot es "Kaxan", que en dialecto Maya quiere decir "el buscador", la figura N° 1 muestra una foto y el dibujo CAD del ROV. Las características de Kaxan son: dimensiones de 1.1m de largo por 0.8m de ancho y 0.65m de alto; peso aproximado de 80Kg; sensores de profundidad, posición, temperatura, orientación, humedad; 4 propulsores de corriente directa sin escobillas que dan al robot 4 grados de libertad, dos lámparas de alógeno, cámara submarina LED multi SeaCam modelo 2065 y un brazo manipulado de 2 grados de libertad más una pinza. Todo ello le permite, al ROV, ser empleado en el uso de inspecciones (en aguas sumerias), a tuberías y estructuras que estén localizadas en profundidades menores a 120 m (máxima profundidad de trabajo del robot).

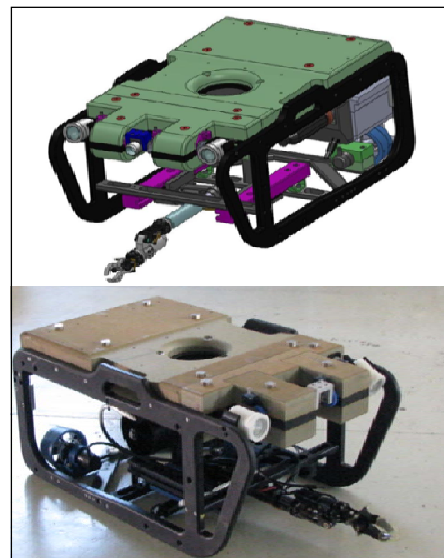


Fig. 1. Robot Submarino ROV "KAXAN".

Para generar movimiento y dirección, el ROV usará 4 propulsores como los de la figura N° 2 (proveedor Tecnam). Los propulsores tienen las siguientes características: voltaje de alimentación 150vcd a 3A; voltaje de control para el driver +/- 5vcd; codificador incremental interno de 32 pulsos por revolución; profundidad de operación estándar de 750m, empuje máximo de 23lbf hacia adelante y 12lbf en reversa.



Fig. 2. DC Brushless Thruster Tecnam, model 520.

El presente trabajo se centra en la identificación de una función de transferencia que modele el comportamiento del propulsor con la finalidad de que, esta función de transferencia o modelo matemático, sirva para poder implementar un control PID de velocidad de giro al propulsor, tanto en simulación como en tiempo real. Para realizar la identificación y control, se armará un circuito que controle el encendido y velocidad del propulsor; por medio de una tarjeta de adquisición de datos (National Instruments) y un programa desarrollado en LabVIEW.

La metodología de operación será la siguiente. Se aplicará un voltaje de excitación al propulsor, entre 3.0 y 3.5 volts (voltaje medio de control del propulsor) el cual generará, por medio del encoder, pulsos con respecto al tiempo, que son convertidos a velocidad angular (ω) mediante la siguiente ecuación:

$$\omega = \frac{\Delta\theta}{\Delta t} \quad (1)$$

$$\Delta\theta = \frac{p \cdot 2 \cdot \pi}{\text{Nr. de pulsos por revolución del encoder}} \quad (2)$$

donde p es el número de pulsos censados y Δt el tiempo de muestreo. Los datos generados se almacenarán en un archivo.

La función será determinada por un método numérico conocido como Mínimos Cuadrados; que es

una técnica en la que dados un conjunto de pares coordenados, se intenta encontrar la función que mejor se aproxime a los datos, de acuerdo con el criterio de mínimo error cuadrático; así que el archivo generado, se procesará en el algoritmo de mínimos cuadrados dentro del ambiente de Matlab de donde se obtendrá una función de transferencia y un gráfico de la aproximación de los datos experimentales. La metodología propuesta es fuera de línea, por ser la más adecuada y simple para este proceso.

El control PID continuo en el tiempo que se aplicará será simulado y sintonizado en Matlab/Simulink. Éste servirá como elemento para tener un panorama de cómo será la señal del PID discreto, que se programará en LabVIEW para controlar el propulsor en tiempo real.

2. Mínimos Cuadrados

En esta sección se describe la metodología de Mínimos Cuadrados por ser este el método usado para la identificación del propulsor.

El principio de los Mínimos Cuadrados indica que los parámetros de un modelo se deben elegir de tal forma que: "La suma de los cuadrados de las diferencias entre los valores de la salida observados (reales) y los estimados, multiplicada por factores que midan el grado de precisión sea un mínimo"[3,6]. Así pues, considerando un modelo genérico:

$$G(z^{-1}) = \frac{b_0 + b_1 z^{-1} + \dots + b_m z^{-m}}{1 + a_1 z^{-1} + \dots + a_n z^{-n}} \cdot z^{-d} \quad (3)$$

La identificación del mismo consiste en la estimación de los parámetros desconocidos del sistema: $a_1, a_2, \dots, a_n, b_0, b_1, \dots, b_m$. Independientemente de que el sistema original $G(z^{-1})$ puede ser continuo, para describir el comportamiento dinámico de un conjunto de mediciones de su entrada y su salida, tomadas durante un cierto número de intervalos de tiempo T , es posible asumir el siguiente modelo lineal:

$$y(k) = \sum_{j=1}^n a_j y(k-j) + \sum_{j=0}^m b_j u(k-j) + e(k) \quad (4)$$

El cual simplificado analíticamente se puede representar como:

$$y(k) = F^T z(k) + e(k) \quad (5)$$

donde \mathbf{F}^T y $\mathbf{z}(k)$ son respectivamente los vectores de los parámetros de las mediciones. Ambos vectores pueden definirse como:

$$\mathbf{F}^T = [b_0 a_1 b_1, \dots, a_n b_n] \quad (6)$$

$$\mathbf{z}^T(k) = \begin{bmatrix} u(k)y(k-1)u(k-1), \dots \\ \dots, y(k-n)u(k-n) \end{bmatrix} \quad (7)$$

Así, resolviendo este producto matricial para la pseudo-inversa (8), se obtendrían los parámetros necesarios que mejor ajusten los puntos planteados inicialmente los cuales forman parte la función de transferencia (3).

$$\hat{\mathbf{F}}^T(z) = (\mathbf{Z}^T(z) * \mathbf{Z}(z))^{-1} (\mathbf{Z}^T(z) * \mathbf{Y}(z)) \quad (8)$$

3. Control PID

Para controlar la velocidad del propulsor se usará un control PID (control Proporcional Integral Derivativo), que es un mecanismo de control por realimentación. Ajustando las tres constantes del algoritmo de control PID, el controlador puede proveer un control diseñado para lo que requiera el proceso a realizar [4]. La respuesta del controlador puede ser descrita en términos de respuesta del control ante un error, el grado al cual el controlador llega al "set point", y el grado de oscilación del sistema. El algoritmo de cálculo del control PID se da en tres parámetros distintos: el proporcional, el integral, y el derivativo. El valor Proporcional determina la reacción del error actual. El Integral genera una corrección proporcional a la integral del error, esto nos asegura que aplicando un esfuerzo de control suficiente, el error de seguimiento se reduce a cero. El Derivativo determina la reacción del tiempo en el que el error se produce. La suma de estas tres acciones es usada para ajustar al proceso vía un elemento de control. Dentro del PID continuo la fórmula del proporcional está dada por:

$$P_{sal} = K_p e(t) \quad (9)$$

La fórmula del integral está dada por:

$$I_{sal} = K_i \int_0^t e(\tau) d\tau \quad (10)$$

La fórmula del derivativo está dada por:

$$D_{sal} = K_d \frac{de}{dt} \quad (11)$$

donde $e(t)$ es el error de seguimiento, K_p es la ganancia proporcional, K_i la ganancia de la acción integral, K_d la ganancia de la acción derivativa. Por otra parte un PID discreto [1] se define por la siguiente fórmula:

$$u(kT) = K \left\{ e(kT) + \frac{1}{T_i} \left[\frac{e(0)+e(T)}{2} + \frac{e(T)+e(2T)}{2} + \dots + \frac{e((k-1)T)+e(kT)}{2} \right] + T_d \frac{e(kT)-e((k-1)T)}{T} \right\} \quad (12)$$

Donde; $e(kT)$ es el error en el tiempo (kT), K es la ganancia proporcional, T_i es el tiempo integral, T_d el tiempo derivativo y $u(kT)$ es la salida del control en el tiempo (kT).

4. Electrónica de Instrumentación

La electrónica utilizada para la identificación consta de una tarjeta aisladora de potencia modelo ISO-4 (recomendado por el proveedor, ver figura N° 3), cuyas características son mostradas en la tabla 1.

Input	Output
(4) non-isolated +/-5v analog control signals	(4) fully isolated +/-5v analog control signals
12vdc, 1.5A. non-isolated instrumentation power	(4) fully isolated 12vdc, 150mA instrumentation power supplies

Tabla. 1. Especificaciones ISO-4

Debido a que el agua causa a la señal del encoder del propulsor cierta cantidad de ruido, se propuso utilizar un filtro basado en el circuito integrado MC74HC14A que suprime el ruido y se logran tener pulsos totalmente cuadrados.



Fig. 3. Analog Isolation Card ISO-4.

Como ya se mencionó, la señal generada por los pulsos del encoder es captada por una tarjeta de adquisición de datos, DAQ de National Instruments (modelo USB - 6211), posteriormente es procesada y almacenada (vía conexión USB) en la PC mediante un programa desarrollado en LabVIEW. El diagrama de bloques de la electrónica de instrumentación se muestra en la figura N° 4.

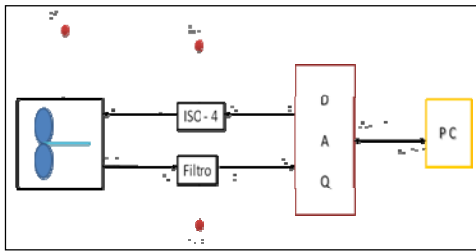


Fig. 4. Diagrama de conexiones del circuito de control para propulsor

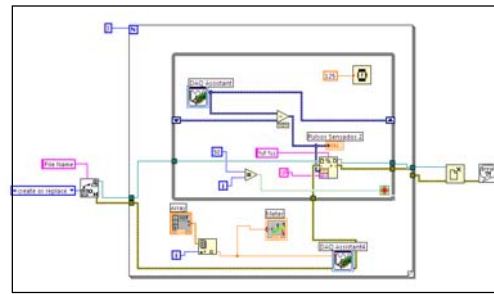


Fig. 6. Interfaz de LabVIEW (Diagrama de conexión).

5. Software de Instrumentación y Control

Esta sección está compuesta por dos programas: 1) Programa de Identificación del propulsor (LabVIEW y Matlab); 2) Programa de control en tiempo real (Matlab y LabVIEW). Los cuales son descritos a continuación.

La interfaz realizada en LabVIEW para la lectura y almacenamiento de los pulsos generados (velocidad del propulsor) se observa en la figura N° 5 y 6. Ésta permite variar el tiempo de muestreo del programa de adquisición y número de muestras deseadas [5]. Asimismo para atenuar la señal, se aplica (por software) un filtro promedio móvil que disminuye el ruido originado por el medio. De igual manera (en LabVIEW) es programado un PID discretizado para control de velocidad de giro (ver figura N° 9). Este programa permite control y monitorear gráficamente la velocidad del propulsor en revoluciones por minuto, así como cambiar el valor de las ganancias del control proporcional, integral y derivativo con el fin de sintonizar de manera heurística el tiempo de respuesta del propulsor así como la disminución del error.

Por otra parte, en MATLAB se programó el algoritmo de Mínimos Cuadrados para obtener una función de transferencia, que modela el comportamiento del propulsor a partir de datos experimentales, el código del programa se muestra en la figura N° 7. La función obtenida, servirá posteriormente para poder simular en Matlab/Simulink un control PID continuo (ver figura N° 8).

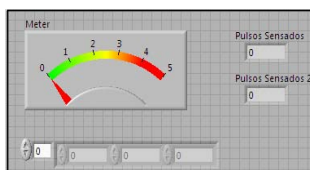


Fig. 5. Interfaz de LabVIEW (Panel frontal).

```

%METODO DE MINIMOS CUADRADOS FUERA DE LINEA PARA
IDENTIFICACION DE PLANTAS
clc;
clear all;
close all;
format short

uu=1;
nn=1;
ar=fopen('data2.txt','r'); %lectura d datos archivo
datos=fscanf(ar,'%f',[2,206]); %asignación
de valores de archivo a variable

fclose(ar);
u=(datos(2,:));%asignación d valores del archivo
y=(datos(1,:)); %asignación d valores del archivo
n=2;
t=length(y);
T=1:1:t;

%Generación de matriz Z
Z(1,:)=u;
for i=1:n
    for k=1:t-i
        Z(2*i,k+i) = y(k);
        Z(2*i+1,k+i) = u(k);
    end
end

%Calculo del estimador con mínimos cuadrados
P = inv(Z'*Z')*(Z'*y')

%Simulación
n2_1=2*n+1;
z=zeros(1,n2_1)';

for k=1:t
    for i=n2_1-2:-1:1
        z(i+2)=z(i);
    end
    z(1)=u(k);
    if k>1
        z(2)=ye(k-1);
    end
    ye(k)=P'*z;
end

%Asignación de valores a variables numerador
y denominador
lim=size(P);
for u=1:2:lim
    num(uu)=P(u);
    uu=uu+1;
end
for u=2:2:lim
    den(nn)=P(u);
    nn=nn+1;
end

num
den
%ploteo de la simulación y aproximación
figure(2)
plot(T,u,'b',T,y,'g',T,ye,'r')
grid on;
    
```

Fig. 7. Código del Algoritmo de mínimos cuadrados aplicado en Matlab.

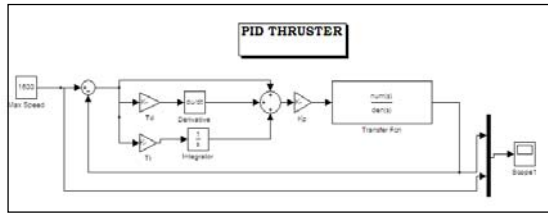


Fig.8. PID continuo diseñado en Matlab/Simulink para control del propulsor (en simulación).

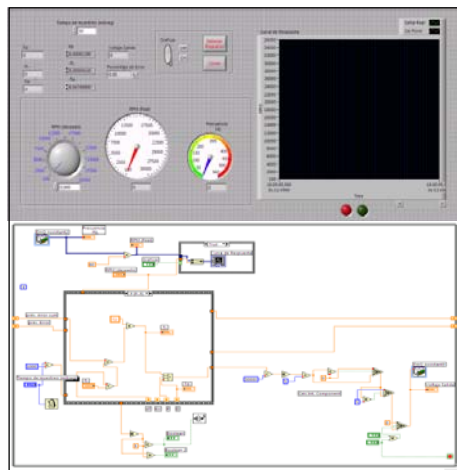


Fig.9. Aplicación en LabVIEW de control PID discreto para el propulsor en tiempo real (Panel frontal y diagrama de conexión).

6. Experimentación

Esta sección describe la metodología usada para la identificación del propulsor y el comportamiento que tuvo el control de velocidad de giro del propulsor.

6.1 Resultados de la Identificación

El experimento consiste en poner dentro de un tonel (o barril) con agua el propulsor, a continuación, inducir un voltaje de excitación entre 3.0 y 3.5 volts, (debido a que éste es el voltaje medio de operación respecto del voltaje máximo nominal) durante 500 milisegundos, se toman lecturas de los pulsos generados cada 10 milisegundos utilizando la DAQ (ver figura N° 10). Al finalizar el tiempo, se detiene el propulsor y se guardan los datos mediante el programa que fue desarrollado en LabVIEW. Posteriormente los datos archivados se llaman desde el algoritmo de mínimos cuadrados en Matlab, en el cual, se tiene como variable el orden de la

aproximación que se desee realizar, en este caso se seleccionó que fuera de segundo orden.

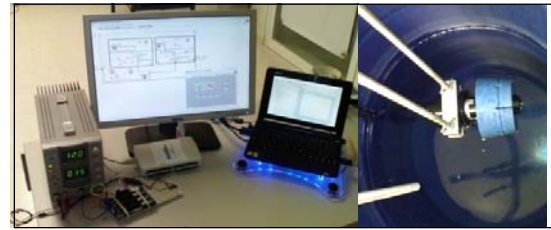


Fig. 10. Experimentos en el Laboratorio

Finalmente el programa de Matlab toma estos datos experimentales y obtiene los parámetros de la función de transferencia planteada en (3) que mejor se aproximan. El programa arroja una gráfica (ver la figura N° 11) de aproximación donde muestran los datos obtenidos en el experimento y la gráfica de la función de transferencia obtenida con Mínimos Cuadrados.

Al ejecutar el experimento, se obtuvieron 50 puntos, con un tiempo de muestreo de 10 milisegundos. La función de transferencia discretizada de segundo orden obteniendo es la siguiente:

$$G(z) = \frac{04.20}{2.026 z^2 - 0.788 z - 1} \quad (12)$$

Y su equivalente en tiempo continuo es la siguiente función:

$$G(s) = \frac{406.6 s^2 - 6.667 e004 s + 2.672 e006}{s^2 + 28.84 s^2 + 41.94 s + 1.309 e004} \quad (13)$$



Fig. 11. Gráfico de la aproximación por mínimos cuadrados

6.2 Resultados del Control

Para el control PID discreto programado en LabVIEW se sintonizo por el método heurístico y se obtuvieron las siguientes ganancias:

$$K_p = 850 \times 10^{-7}$$

$$T_i = 515 \times 10^{-3}$$

$$T_d = 15.57 \times 10^{-3}$$

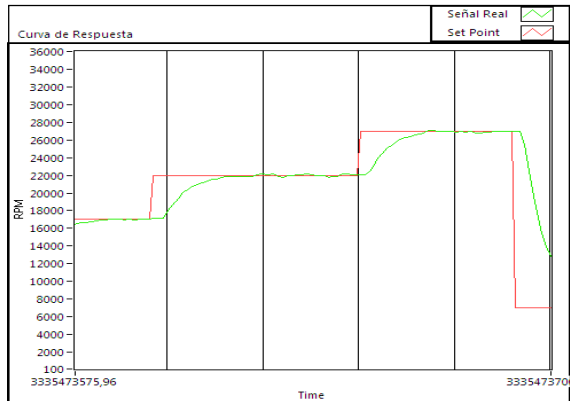


Fig. 12. Gráfico de control PID aplicado al propulsor

Las pruebas de control realizadas al propulsor fueron aplicando diversas velocidades. En la figura N° 12 se muestra el comportamiento del propulsor cuando a este se le piden velocidades de referencia de: 17000rpm, 22000rpm y 27000 rpm.

7. Conclusiones

En el transcurso del trabajo se pudo observar cómo un método numérico simple como Mínimos Cuadrados, complementado con una herramienta de control como el PID, fue un resultado poderoso para poder identificar y controlar las velocidades de giro de un propulsor empleado para ROVs. La función de transferencia, obtenida de manera experimental y que modela al propulsor, sirve perfectamente como base para generar un panorama de cómo será el comportamiento de la señal del PID discreto.

8.1 Trabajos Futuros

En un futuro se pretende trabajar con 4 propulsores, operados simultáneamente desde una consola de control, montados sobre el robot submarino (ROV); para lo cual será de vital importancia poder controlar con precisión su velocidad, ya que estos determinarán la dirección, posición, orientación y profundidad de todo el sistema. Cabe mencionar que en esta ocasión, la retroalimentación empleada para control fue tomada del codificador del propulsor, pero para el control de

todo el sistema ROV además del codificador, se controlará con otro tipo de retroalimentación basada en un sensor de profundidad, el cual, proporcionará la distancia que el ROV mantiene con respecto a la superficie del agua.

Agradecimientos

Este trabajo fue financiado gracias a los proyectos:
 1) Ciencias Básicas CONACYT 2005, número 26032.
 2) Proyecto interno de CIDESI, con clave QI0037.

Referencias

- [1] Katsuhiko O. “*Sistemas de Control en Tiempo Discreto*”, Prentice Hall, México, segunda edición, 1996.
- [2] Fossen T. “*Guidance and Control of Ocean Vehicles*”, John Wiley & Sons Ed, Canada, primera edición, 1994.
- [3] Chapra Steven. “*Métodos Numéricos para Ingenieros con Programas y Aplicaciones*”, Mc Graw Hill, México, quinta edición, 2007.
- [4] Katsuhiko O. “*Ingeniería de Control Moderna*”, Prentice Hall, México, tercera edición, 1998.
- [5] Lajara J, Pelegri J. “*LabVIEW: entorno gráfico de programación*”, Marcombo, México, primera edición, 2007.
- [6] Ronquillo G, Herrera G. “*Brushless DC Digital Servo Motor Control*”, Proceedings of MUSME 2002, the International Symposium on Multibody Systems and Mechatronics. Mexico City, 12-14 September 2002.
- [7] Gomes R, Sousa J, Lobo F. “*Modeling and Control of the (IES) Project ROV*”, IEE European Control Conference, Engineering Faculty of Porto University, FEUP, 2003.