

# Implementation of an Arrangement of Thrusters on a Robot for Confined Spaces.

Rony Caballero, Dr.<sup>1</sup>, Alberto Cogley, Ing.<sup>1</sup>, Adan Vega, Dr.<sup>1</sup>

<sup>1</sup> Universidad Tecnológica de Panamá, Panamá, rony.caballero@utp.ac.pa; alberto.cogley@utp.ac.pa; adan.vega@utp.ac.pa

*Abstract— The application of engines in confined spaces requires configurations and algorithms to properly manage the effects of saturation. These nonlinear effects depend on the distance to walls and the water surface. This paper presents an algorithm for real-time address these effects based on redundant configurations are proposed boosters.*

*Keywords— Robotics, underwater, saturation, pseudoinverse*

**Digital Object Identifier (DOI):** <http://dx.doi.org/10.18687/LACCEI2015.1.1.211>

**ISBN:** 13 978-0-9822896-8-6

**ISSN:** 2414-6668

**13<sup>th</sup> LACCEI Annual International Conference:** “Engineering Education Facing the Grand Challenges, What Are We Doing?”  
July 29-31, 2015, Santo Domingo, Dominican Republic      **ISBN:** 13 978-0-9822896-8-6      **ISSN:** 2414-6668  
**DOI:** <http://dx.doi.org/10.18687/LACCEI2015.1.1.211>

# Implementación de un arreglo de propulsores en un robot para espacios confinados.

Rony Caballero, Dr., Alberto Cogley, Ing. Y Adan Vega, Dr.

Universidad Tecnológica de Panamá, Panamá, rony.caballero@utp.ac.pa; alberto.cogley@utp.ac.pa; adan.vega@utp.ac.pa

**Resumen—** La aplicación de propulsores en espacios confinados requiere de configuraciones y algoritmos que permitan manejar apropiadamente los efectos de la saturación. Estos efectos no lineales dependen de la distancia a las paredes y a la superficie del agua. En este trabajo se propone un algoritmo que permita abordar en tiempo real estos efectos en configuraciones basadas en propulsores redundantes.

**Palabras clave—**Robótica, submarino, saturación, pseudoinversa

## I. INTRODUCCIÓN

La aplicación de vehículos submarinos se ha extendido a múltiples aplicaciones [1][2][3]. Sin embargo, la mayoría de éstas se ha limitado a su utilización a aguas abiertas. Por otro lado, resulta de interés la utilización de robots o vehículos operados remotamente para labores de inspección dentro de alcantarillas y otros espacios confinados. En estos casos, la mayoría de las realizaciones clásicas presentan limitaciones importantes, ya que estos no presentan—radios de giro lo suficientemente pequeños y el arreglo de propulsores utilizado usualmente limita la maniobrabilidad en estos entornos confinados [4].

Una forma de solucionar este problema se logra utilizando arquitecturas que utilizan propulsores redundantes. En estos casos no solo se puede obtener una excelente maniobrabilidad, sino que también se puede optimizar mejor la potencia demandada entre los propulsores. No obstante, es importante destacar que muchos de los propulsores disponibles en el mercado presentan limitaciones no solo en los valores máximos de fuerza suministrada, sino también en la tasa de cambio de la misma. Estos valores máximos cambian en función de la distancia de los propulsores a las paredes y a la superficie del agua.

Para resolver estos problemas, algunas investigaciones han propuesto, aplicar algunos algoritmos que minimizan la norma infinito en los propulsores [5][6]. Sin embargo, se ha demostrado que estos algoritmos no garantizan que la limitación de la tasa de cambio pueda cumplirse [7]. Para resolver este problema, otros autores proponen la utilización de algoritmos que llevan a la saturación de las consignas de los propulsores en el espacio nulo [7][8]. Si bien es cierto, este enfoque resulta muy elegante, presenta dificultades para implementarlo en tiempo real en la computadora a bordo del submarino en espacios confinados. Esto se debe a que la fuerza que puede proporcionar cada propulsor es función del sentido de la fuerza, de la distancia a las paredes y la distancia

a la superficie del agua, lo que aumenta la complejidad del algoritmo.

En este trabajo se propone la utilización de un algoritmo que permita utilizar los propulsores en entornos confinados, manteniendo las consignas de los propulsores dentro del rango de operación para los que fueron diseñados.

## II. CONFIGURACIÓN Y CARACTERIZACIÓN DE PROPULSORES

### A. Configuración para espacios confinados

El vehículo submarino de arquitectura ODIN resulta apropiado para la inspección de espacios confinados (Véase Fig. 1). Esta configuración consta de 8 propulsores ubicados de manera que puedan obtenerse 6 grados de libertad (Véase Fig. 2). La localización de cada propulsor respecto a un eje de coordenadas ubicado en el centro geométrico del vehículo puede parametrizarse así:

- $a$  es la distancia de tanto el propulsor 1 como del 2, 3 y 4 al plano  $Z=0$
- $b$  es la distancia de tanto el propulsor 1 como del 2, 3 y 4 al plano  $Y=0$
- $c$  es la distancia de tanto el propulsor 5 como del 7 al plano  $Z=0$
- $d$  es la distancia de tanto el propulsor 6 como del 8 al plano  $Y=0$



Fig. 1 Vehículo submarino para inspección de espacios confinados

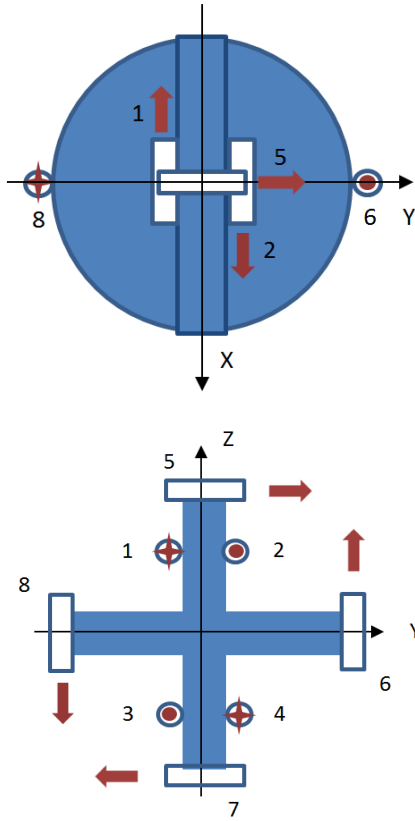


Fig. 2 Configuración propuesta de propulsores para espacios confinados.

Es importante destacar que para cada propulsor se coloca uno en contraposición. Esto permite compensar mejor las no linealidades de los mismos y conservar la alta maniobrabilidad del submarino. En consecuencia, el vector de pares y fuerzas generados puede representarse en función de las fuerzas generadas en los ocho propulsores mediante la relación (1).

$$F = JT \quad (1)$$

Donde,  $J$  es una matriz de transformación que relaciona ambos vectores, definida por:

$$J = \begin{bmatrix} -1 & 1 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 & -c & d & -c & d \\ -a & a & -a & a & 0 & 0 & 0 & 0 \\ -b & -b & b & b & 0 & 0 & 0 & 0 \end{bmatrix} \quad (2)$$

$T = [T_1 \ T_2 \ T_3 \ T_4 \ T_5 \ T_6 \ T_7 \ T_8]^T$  es el vector de fuerzas proveniente de cada propulsor y  $F = [F_x \ F_y \ F_z \ M_x \ M_y \ M_z]^T$  es el vector de fuerzas generalizados en el vehículo.

### B. Caracterización de los propulsores

Los propulsores utilizados en robots submarinos son usualmente controlados por una señal de voltaje modulada en ancho de pulso. Sin embargo, la fuerza que puede proveer cada propulsor no solo es función del voltaje en terminales, sino también del sentido de la fuerza que genera cada propulsor. Esto se debe a que la eficiencia del mismo no necesariamente es simétrica. Además, puede presentarse zonas muertas y de saturación (véase Fig. 3).

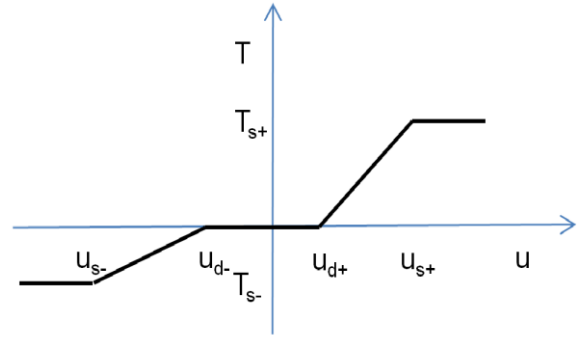


Fig. 3 Relación entre la fuerza generada por cada propulsor y el voltaje en terminales.

En consecuencia, la relación entre el voltaje en terminales y la fuerza de propulsión puede representarse de la siguiente manera:

$$T(u) = \begin{cases} T_{s-} & u \leq u_{s-} \\ A_- u & u_{s-} < u < u_{d-} \\ 0 & u_{d-} < u < u_{d+} \\ A_+ u & u_{d+} < u < u_{s+} \\ T_{s+} & u \geq u_{s+} \end{cases} \quad (3a)$$

Sujeto a las restricciones,

$$\begin{aligned} -T_{i\max} &\leq T_{is-} \leq T_i \leq T_{is+} \leq T_{i\max} \\ \left| \frac{dT_i}{dt} \right| &\leq r_{iT} \end{aligned} \quad (3b) \text{ y } (3c)$$

Otros efectos importantes a considerar son la distancia de cada propulsor a las paredes y la superficie del agua (véase Fig. 3). La influencia de estos factores se evidencia experimentalmente. Así, la cercanía a las paredes usualmente se manifiesta por cambios en el parámetro  $A_+$  o  $A_-$  del propulsor, mientras que la cercanía a la superficie usualmente produce un cambio en los niveles de saturación  $T_{s+}$  o  $T_{s-}$  (véase Fig. 4).

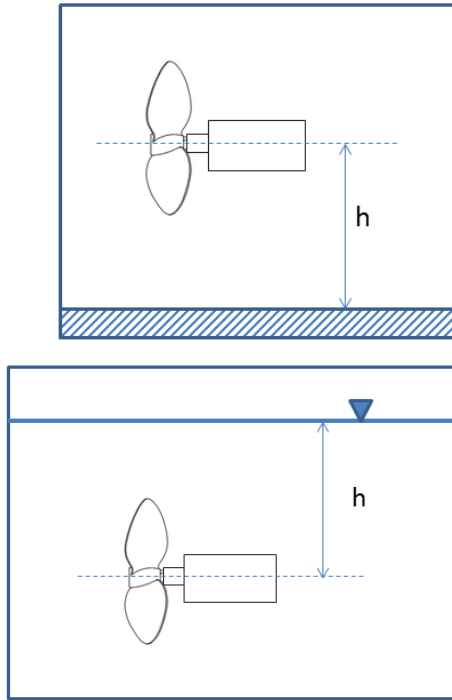


Fig. 4. Efecto de la distancia a las paredes y distancia a la superficie del agua

### III. GENERACIÓN DEL VECTOR DE FUERZAS PROVISTAS POR LOS PROPULSORES

Para determinar la fuerza que debe suministrar cada propulsor se consideran los efectos anteriormente descritos en el apartado anterior. En este sentido, existen varias alternativas para obtener el vector de fuerzas  $T$  proveniente de los propulsores.

#### A. Algoritmos basados en la minimización de la norma infinito

Para aplicar estos algoritmos [Ha and Lee, 2002] se debe minimizar:

$$\|T\|_{\infty} = \max \left[ \left| \frac{T_1 - T_{1m}}{T_{1p}} \right|, \left| \frac{T_2 - T_{2m}}{T_{2p}} \right|, \dots, \left| \frac{T_8 - T_{8m}}{T_{8p}} \right| \right] \quad (4)$$

$$\text{Donde, } T_{im} = \frac{1}{2}(T_{is+} + T_{s-}) \text{ y } T_{ip} = \frac{1}{2}(T_{is+} - T_{s-})$$

Sin embargo, esta opción no garantiza la restricción,

$$\left| \frac{dT_i}{dt} \right| \leq r_{iT} \quad (5)$$

Para solventar este problema, se puede aplicar algoritmos basados en programación lineal [6] o programación cuadrática. Pero estos métodos pueden generar un alto costo computacional para ejecuciones en tiempo real.

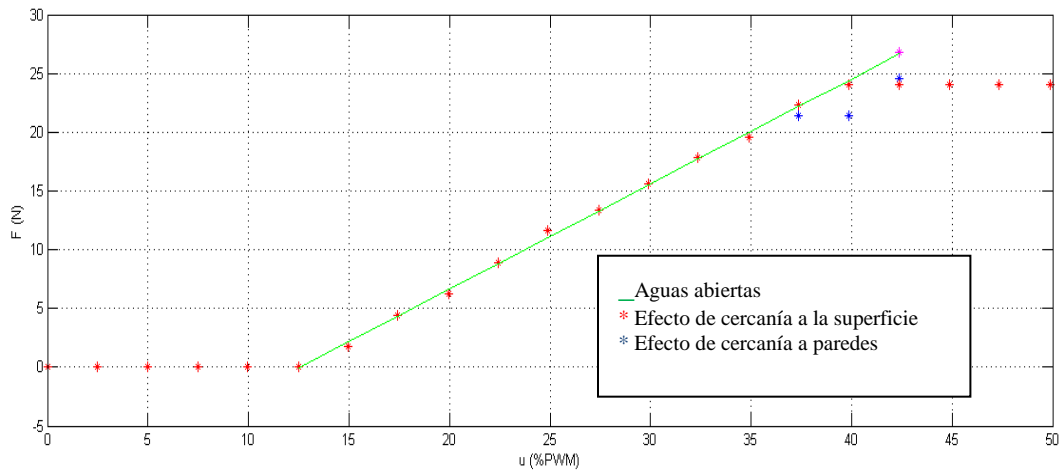


Fig. 5. Efecto de la distancia a las paredes y distancia a la superficie del agua sobre la fuerza generada en el propulsor.

### B. Algoritmos basados en la aplicación de la pseudoinversa generalizada

Estos algoritmos buscan explotar las propiedades de la ecuación:

$$T = J^+ F + N\phi \quad (6)$$

Donde,  $J^+ = J^T [JJ^T]^{-1}$  es la parte de la solución que minimiza la norma 2 y  $N = I - J^T [JJ^T]^{-1} J$  corresponde a la proyección de  $\phi$  en el espacio nulo de  $J$ . En este sentido algunos autores han propuesto abordar las restricciones aplicando algoritmos de saturación [7] [8] en el espacio nulo de  $J$ . Sin embargo, estos algoritmos no son fáciles de generalizar para todos los casos.

### C. Aplicación de la pseudoinversa generalizada con transformación lineal

En este trabajo, se propone otra alternativa para abordar las restricciones. Primero, resulta conveniente aplicar la transformación lineal,

$$T = \rho R T' + T_0 \quad (7)$$

Donde,  $R = \text{diag}(R_1, R_2, \dots, R_n)$ ,  $R_i$  es una constante que puede elegirse convenientemente,  $\rho = \text{diag}(\rho_1, \rho_2, \dots, \rho_n)$   $\rho_i$  es una variable de conmutación que puede tomar un valor de 1 o 0 y  $T_0'$  es un vector arbitrario constante. Por lo tanto,

$$F = J \rho R T' + J T_0 \quad (8)$$

Sujeto a las restricciones,

$$T_{is-} \leq T_i \leq T_{is+} \quad (9a)$$

$$\left| \frac{dT_i}{dt} \right| \leq r_{iT} \quad (9b)$$

Luego, si se aplica la pseudoinversa generalizada para obtener,

$$T' = [J \rho R]^+ [F - J T_0] + N\phi \quad (10)$$

Donde,  $[J \rho R]^+ = R \rho J^T [J \rho R^2 \rho J^T]^{-1}$  y

$N = I - [J \rho R]^+ J \rho R$  corresponde a la proyección de  $\phi$  en el espacio nulo de  $J \rho R$ . Por lo tanto, se puede demostrar que,

$$T = \rho R [J \rho R]^+ [F - J T_0] + \rho R N \phi + T_0 \quad (11)$$

Además, si  $T_0' = R^{-1} T_0$  y si todo  $\rho_i = 1$ , entonces:

$$T = R [J R]^+ F + R N (T_0' + \phi) \quad (12)$$

Así,  $T_0'$  se proyecta en el espacio nulo de  $J R$ . En consecuencia, se puede aprovechar la estructura de (11) de la siguiente manera:

- Se puede elegir  $T_0$  de modo que este cerca del valor de vector de fuerzas anterior proyectando  $T_0' = R^{-1} T_0$  en el espacio nulo de  $J R$ . Luego, se aplica (11) para corregir los valores de  $T$ . Esto permite una respuesta suave en los propulsores ayudando a cumplir la restricción (9b).
- Si se incumple la restricción (9a) se puede elegir  $T_{i0}$  de modo que tome el valor límite que  $T_i$  pueda tomar y luego se desactiva el grado de libertad haciendo  $\rho_i = 0$ . Para volver a calcular (11).
- Es importante destacar que la pseudoinversa  $[J \rho R]^+$  existirá si  $\det(J \rho R^2 \rho J^T) \neq 0$ .

## IV. DETERMINACIÓN DEL VECTOR DE FUERZAS DE LOS PROPULSORES SIN INVERSIÓN DE MATRICES

Los algoritmos convencionales basados en la utilización directa de la matriz de pseudoinversa, no solo pueden presentar problemas de mal condicionamiento, sino también pueden provocar una importante carga computacional para la computadora de control a bordo del robot. Esto se debe principalmente a que el tiempo que toma la inversión de matrices puede ser importante. Sin embargo, es conocido que la inversión de matrices puede evitarse si se utilizan algoritmos basados en retroalimentación [9]. En este sentido es posible demostrar que (11) para  $\phi = 0$  puede aproximarse por:

$$T = \lim_{q \rightarrow \infty} \rho R^2 \rho J^T \left[ J \rho R^2 \rho J^T + q^{-1} I \right]^{-1} [F - J T_0] + T_0 \quad (13)$$

Si se aplica la traspuesta a (13) se obtiene,

$$T^T = \lim_{q \rightarrow \infty} [F - J T_0]^T \left[ q J \rho R^2 \rho J^T + I \right]^{-1} q J \rho R^2 \rho + T_0^T \quad (14)$$

Si se define,

$$\xi^T = [F - J T_0]^T - Y^T R \rho J^T \quad (15)$$

$$Y^T = \xi^T q J \rho R \quad (16)$$

$$T^T = Y^T R \rho + T_0^T \quad (17)$$

Así, el sistema descrito por (15), (16) y (17) puede representarse como se muestra en la Fig. 6.

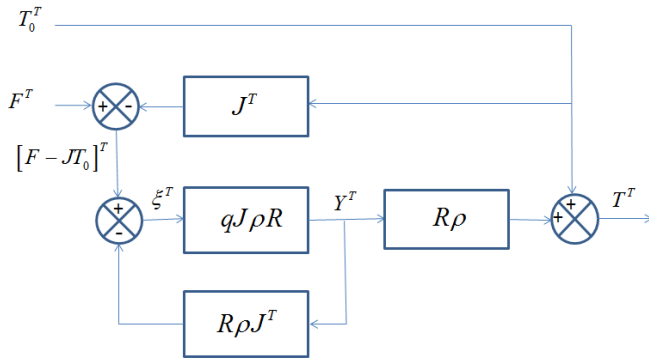


Fig. 6. Modelo del sistema descrito por (15), (16) y (17)

Este modelo tiene la ventaja que evita la utilización de la pseudoinversa gracias a la retroalimentación de alta ganancia. No obstante, el modelo no es causal al implementarse en un algoritmo computacional. Esto se debe a que no se puede realimentar instantáneamente  $Y$ .

Este problema se solventa si  $q$  deja de ser constante y se convierte en un variable cuya magnitud tenga un valor muy grande para frecuencias bajas y un valor bajo para frecuencias altas (véase Fig. 7). Estas características se describen con un integrador que puede aproximarse por,

$$q(z) = \frac{\alpha}{1 - z^{-1}} \quad (18)$$

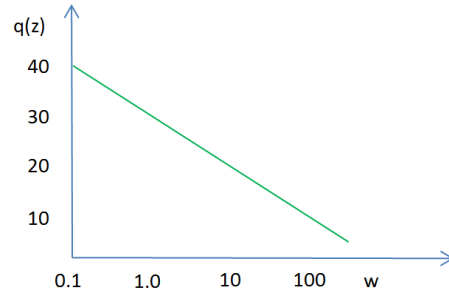
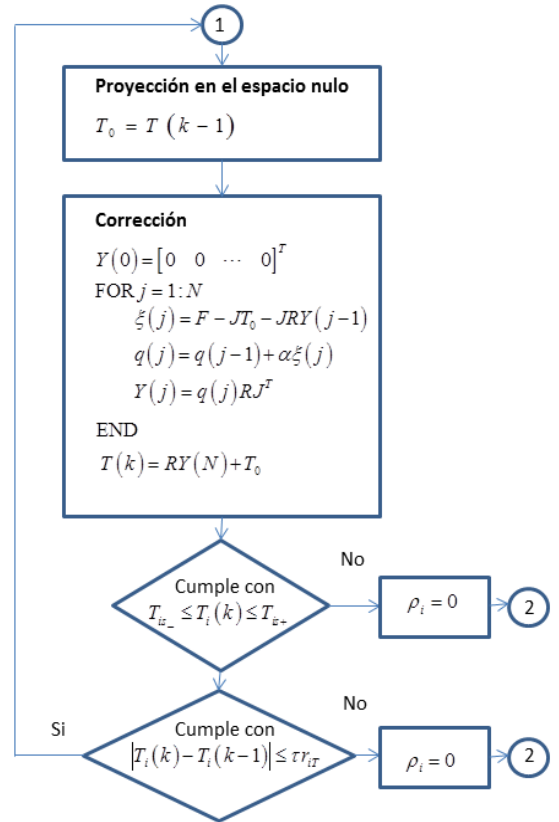


Fig. 7. Diagrama de Bode de  $q(z)$

## V. ALGORITMO PROPUESTO

Como consecuencia del análisis realizado anteriormente, se propone el algoritmo descrito en la Fig. 8 para determinar el vector de fuerzas de los propulsores.



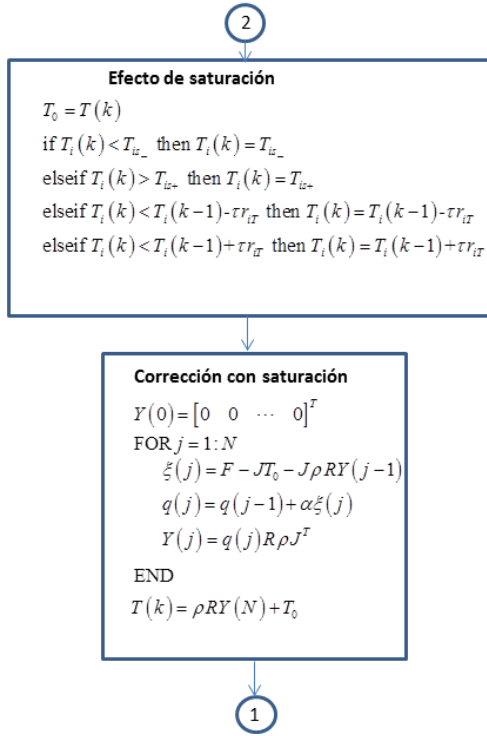


Fig. 8. Algoritmo propuesto para determinar el vector de fuerzas de los propulsores

En este algoritmo se destacan dos eventos importantes. El primero representa al sistema operando sin violar ninguna restricción. Aquí,  $T'_0 = R^{-1}T_0 = R^{-1}T(k-1)$  se proyecta en el espacio nulo de  $JR$ , para luego corregir  $T(k)$  producto de algún cambio en  $JR$  o  $F$ . El segundo evento se activa si se viola alguna de las restricciones. En este caso, se produce el efecto de saturación directamente sobre  $T_0$ , y luego se desactiva el  $\rho_i$  correspondiente. Posteriormente, se corrigen las estimaciones considerando la saturación.

## VI. RESULTADOS BASADOS EN DATA SIMULADA

Para comprobar la efectividad de este algoritmo se evalúan dos casos:

### A. Caso 1

Se tiene un arreglo de 8 propulsores con las restricciones,  $-0.5 \leq T_i \leq 1$  y  $|T_i(k) - T_i(k-1)| \leq 0.5$ . Además, los otros parámetros son:

$$R = \text{diag}([0.75 \ 0.75 \ 0.75 \ 0.75 \ 0.75 \ 0.75])$$

$$T_0 = [-0.4 \ 0.4 \ 0.4 \ 0.2 \ -0.1 \ 0 \ 0 \ 0]^T$$

$$F = [1.70 \ 0.45 \ 0.31 \ -0.08 \ 0.06 \ 0.31]^T$$

$$J = \begin{bmatrix} -1 & 1 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 & -0.3 & 0.3 & -0.3 & 0.3 \\ -0.3 & 0.3 & -0.3 & 0.3 & 0 & 0 & 0 & 0 \\ -0.6 & -0.6 & 0.6 & 0.6 & 0 & 0 & 0 & 0 \end{bmatrix}$$

En este caso, no hay problemas con las restricciones y el algoritmo converge a:

$$T(k) = [-0.45 \ 0.46 \ 0.65 \ -0.10 \ 0.27 \ 0.06 \ -0.18 \ -0.25]^T$$

La rapidez de convergencia puede apreciarse en las Fig. 9 y Fig. 11. Puede apreciarse que se alcanza un error cuadrático medio de 0.01 en 20 iteraciones.

### B. Caso 2

Resulta de interés continuar analizando el caso anterior modificando lo siguiente:

$$T_0 = [-0.45 \ 0.46 \ 0.65 \ -0.10 \ 0.27 \ 0.06 \ -0.18 \ -0.25]^T \text{ y}$$

$F = [2.00 \ 0.45 \ 0.31 \ -0.084 \ 0.06 \ 0.31]^T$ . Esto puede considerarse como un ligero cambio en  $F$  considerando como  $T_0$  la salida del caso anterior.

Ahora, el algoritmo converge a:

$$T(k) = [-0.54 \ 0.56 \ 0.72 \ -0.18 \ 0.27 \ 0.06 \ -0.18 \ -0.25]^T$$

Lo que significa que se viola la restricción,  $-0.5 \leq T_1 \leq 1$ .

Por lo tanto,  $\rho = \text{diag}([0 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1])$  y  $T_0$  se ajusta como,

$$T_0 = [-0.50 \ 0.46 \ 0.65 \ -0.10 \ 0.27 \ 0.06 \ -0.18 \ -0.25]^T$$

Pero entonces, el algoritmo converge a:

$$T(k) = [-0.50 \ 0.60 \ 0.76 \ -0.14 \ 0.27 \ 0.06 \ -0.18 \ -0.25]^T$$

Como puede verse el algoritmo, redistribuye adecuadamente las fuerzas entre los propulsores sin violar las restricciones.

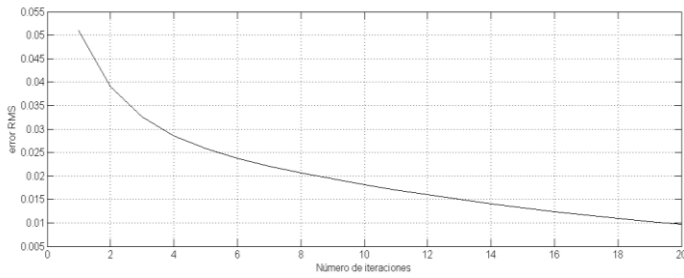


Fig. 9. Evolución del error cuadrático medio en función del número de iteraciones (Caso 1)

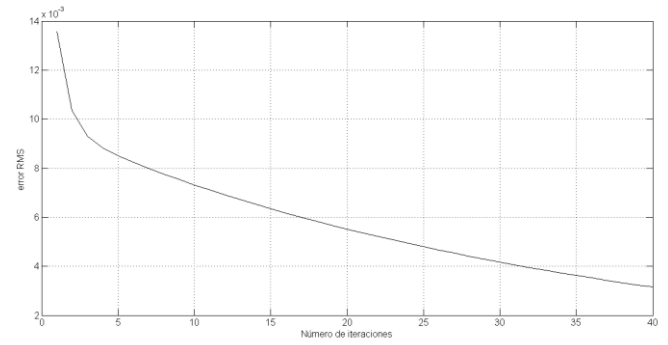


Fig. 10. Evolución del error cuadrático medio en función del número de iteraciones (Caso 2)

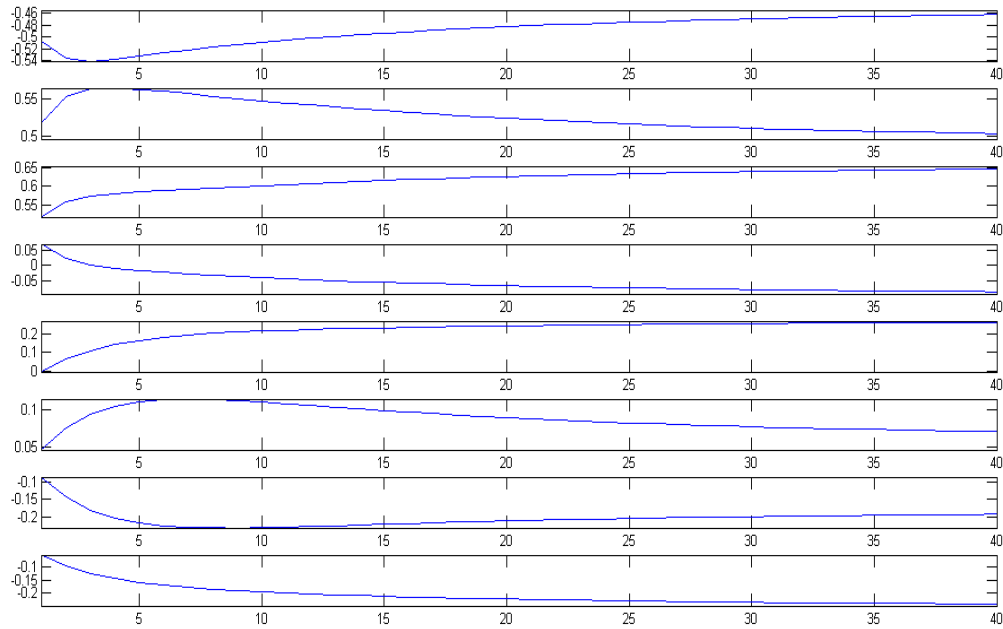


Fig. 11. Evolución y convergencia de los valores de fuerza en los propulsores función del número de iteraciones (Caso 2)



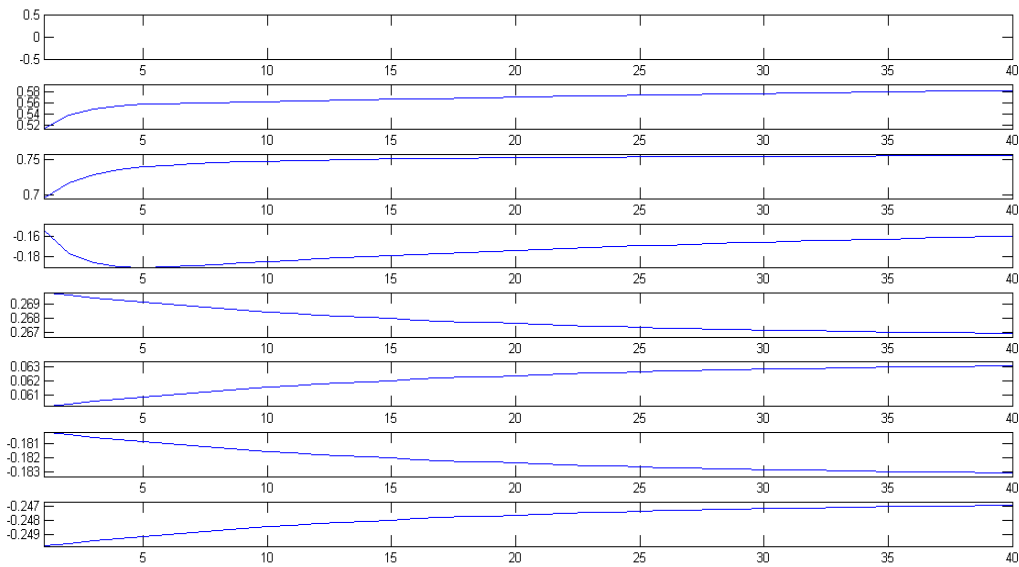


Fig. 12. Evolución y convergencia de los valores de fuerza en los propulsores función del número de iteraciones (Caso 2)

## VII. CONCLUSIONES

En este trabajo se muestra un nuevo algoritmo para la estimación de las fuerzas que deben ser provistas por el arreglo de propulsores. El algoritmo ha demostrado ser efectivo para minimizando en norma 2, pero también considerando las restricciones.

Resulta de interés evaluar este algoritmo en tiempo real. También, podría probarse con pocas modificaciones en aplicaciones relacionadas al cálculo de cinemática inversa.

La utilización de retroalimentación para evitar utilizar matrices inversas ha resultado ser efectivo para evitar problemas de mala condición de las matrices. Además, ha resultado ser muy rápido para la aplicación. Resulta de interés, probar con nuevas estructuras de retroalimentación y control para mejorar todavía la velocidad de convergencia.

## AGRADECIMIENTOS

Los autores desean agradecer a la Secretaría Nacional de Ciencia y Tecnología de Panamá (SENACYT) y a la Universidad Tecnológica de Panamá por el apoyo recibido para esta investigación.

## REFERENCIAS

[1] A. Hanai, K. Rosa, S.K. Choi, J. Yuh. *Experimental Analysis and implementation of Redundant Thrusters for Underwater Robots*.

Proceedings 01 2004 IEEE IRS. International Conference on Intelligent Robots and Systems. September 28 -October 2, 2004. Sendai, Japan.

[2] J Batlle, P Ridao, R Garcia, M Carreras, X Cufí, A El-Fakdi, D Ribas, T Nicosevici, E Batlle, G Oliver, A Ortiz, J Antich. "URIS: Underwater Robotic Intelligent System". Automation for the Maritime Industries 177-203

[3] Choi, H.T. Hanai, A. ; Choi, S.K. ; Yuh, J. "Development of an underwater robot, ODIN-III". International Conference on Intelligent Robots and Systems, 2003. (IROS 2003).

[4] R. Caballero, A Vega, A Berbey, J Rodríguez. "Vehículo submarino de 6 grados de libertad para inspección interna de tuberías". Tenth LACCEI. Panamá City, Panamá, 2012.

[5] Insoo Ha and Jihong Lee. *Analysis on a Minimum Infinity-norm Solution for Kinematically Redundant Manipulators*. ICASE: Institute of Control, Automation, and Systems Engineers, Korea. Vol. 4, No 2, June 2002.

[6] Serdar Soylu, Bradley J. Buckham, Ron P. Podhorodeski. A chattering-free sliding-mode controller for underwater vehicles with fault tolerant infinity-norm thrust allocation. Ocean Engineering 35 (2008) 1647–1659.

[7] Damir Ormcen, Leon Zplajpah, Bojan Nemec. *Compensation of velocity and/or acceleration joint saturation applied to redundant manipulator*. Robotics and Autonomous Systems 55 (2007) 337–344.

[8] Fabrizio Flacco, Alessandro De Luca and Oussama Khatib. *Motion Control of Redundant Robots under Joint Constraints: Saturation in the Null Space*. International Conference on Robotics and Automation River Centre, Saint Paul, Minnesota, USA. May 14-18 2012.

[9] Alexandre N. Pechev Inverse Kinematics without matrix inversion IEEE International Conference on Robotics and Automation Pasadena, CA, USA, May 19-23, 2008