

# Localización de un robot móvil tipo lego con el filtro de Kalman

*Finding a mobile robot lego with the Kalman filter*

Leidy Yolanda López Osorio  
Giovanni Rodrigo Bermúdez Bohórquez

Fecha de recepción: 12 de febrero de 2013

Fecha de aprobación: 30 de abril de 2013

## Resumen

En el proyecto de investigación *Diseño e implementación de agentes robots, actuando de manera cooperativa, en una zona colapsada simulada, como apoyo a organismos de búsqueda y rescate urbano*, se planearon diferentes rutas para la búsqueda y rescate de las víctimas, las cuales eran seguidas por los agentes, presentándose errores de localización debidos a las condiciones físicas de las plataformas. En este artículo se presenta la implementación del filtro de Kalman en una plataforma móvil Lego NXT con el software LabVIEW, para estimar la posición y corregir los errores de localización.

## Palabras clave:

Odometría, estimación, filtro de Kalman, LabVIEW.

\* Universidad Distrital Francisco José de Caldas Grupo de investigación en Robótica Móvil Autónoma Bogotá, Colombia lylopezo@correo.udistrital.edu.co

\*\* Universidad Distrital Francisco José de Caldas Grupo de investigación en Robótica Móvil Autónoma Bogotá, Colombia, gbermudez@udistrital.edu.co

### Abstract

In the investigation project *Design and implementation of robots agents acting cooperatively, in a simulated collapsed environment as organisms support urban search and rescue*, different routes were planned for the search and rescue of victims, which were followed by agents, posing localization errors due to the physical conditions of the platforms.

This article presents the implementation of the Kalman filter on a mobile platform Lego NXT with LabVIEW software to estimate the position and location correct errors.

### Keywords

Odometry, estimation, Kalman filter, LabVIEW.

## 1. Introducción

Conocer la posición y orientación de una plataforma móvil en todo instante, es posible haciendo el análisis matemático de sus características cinemáticas y odométricas; sin embargo, localizar un robot haciendo uso de dicho análisis, es decir utilizando un método incremental, genera inconvenientes debido a la acumulación de errores a lo largo de la trayectoria; haciéndose necesario utilizar un método de corrección que permita ajustar el recorrido de la plataforma de forma periódica. En este caso se implementa la técnica recursiva del filtro de Kalman, que toma el modelo cinemático y las observaciones de un sistema y las transforma en un modelo lineal a través de la expansión de Taylor para actualizar la covarianza de las mediciones [1], con lo que se determinan los parámetros correctos del sistema que evoluciona en el tiempo. El filtro va prediciendo y auto ajustándose con cada nueva medida [2], de esta forma es posible corregir los errores y seguir las rutas establecidas para la búsqueda y el rescate de víctimas correctamente.

El desarrollo de esta técnica recursiva se realiza en LabVIEW, el cual brinda una herra-

menta de estimación y un módulo de programación para el Lego NXT.

## 2. Determinación de parámetros

### 2.1 Modelo cinemático

Figura 1. Plataforma diferencial



Fuente: elaboración propia.

En la figura 1 se muestra la plataforma con tracción diferencial a la cual se le hace el modelo cinemático. El desplazamiento total y la orientación de la plataforma están dados por las ecuaciones (1) y (2) respectivamente, en donde se presentan expresiones generales aplicando todas las constantes correspondientes para la plataforma donde  $x$  y  $\theta$  son el desplazamiento y la orientación de la rueda izquierda, son el desplazamiento y la orientación de la rueda derecha y  $b$  la distancia entre las dos ruedas.

$$\Delta D = \frac{X_{m_L} - X_{m_R}}{2} = 0,0244(\theta_L + \theta_R) \quad (1)$$

$$\Delta \theta = \frac{X_{m_L} - X_{m_R}}{b} = 0,2856(\theta_L - \theta_R) \quad (2)$$

## 2.2 Modelo odométrico

Un sistema que está permanentemente registrando la distancia recorrida describe la evolución temporal de la localización de una plataforma móvil en función de sus propias variables como se muestra en la ecuación (3).

$$X(k+1) = f(X(k), U(k)) + v(k) \quad (3)$$

Donde  $X(k+1)$  es la posición estimada,  $X(k)$  es la posición actual,  $U(k)$  es la entrada del sistema y  $v(k)$  es el vector de errores sistemáticos y no sistemáticos que pueden ser asociados a la plataforma móvil. La entrada del sistema  $U(k)$  se encuentra determinada por la ecuación (4), donde  $\Delta D(k)$  es la distancia recorrida por la plataforma en un intervalo  $(k, k+1)$  y  $\Delta \theta(k)$  es la variación de la orientación en el mismo intervalo.

$$U(k) = [\Delta D(k) \quad \Delta \theta(k)]^T \quad (4)$$

La caracterización del vector de errores de estado  $v(k)$  para la plataforma móvil, es asumido como  $v(k) \approx (0, 0)$ , donde  $v(k)$  es el error característico de estado de la plataforma. Este error se encuentra dado por la ecuación (5); donde puede observarse que los errores no son colineales.

$$Q(k) = \begin{bmatrix} k_D(D(k) \cos \theta(k)) & 0 & 0 \\ 0 & k_D(D(k) \sin \theta(k)) & 0 \\ 0 & 0 & k_{D\theta}(D(k)) + k_\theta(\theta(k)) \end{bmatrix} \quad (5)$$

Siendo el coeficiente de error de traslación de la plataforma relativo a  $\Delta D$  y expresado en  $[^2]$ ; el coeficiente de error de rotación de la plataforma relativo a  $\Delta D$  y expresado en  $[^2]$ ; y el coeficiente de error de rotación de la plataforma relativo a  $\Delta \theta$  y expresado en  $[^2]$ .

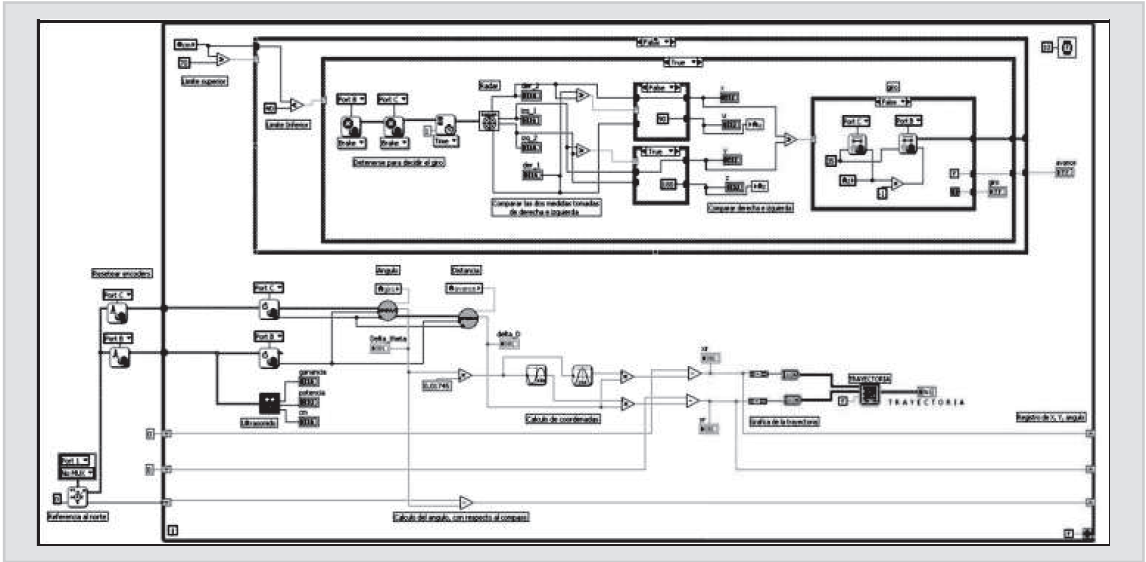
Teniendo en cuenta los procesos de experimentación realizados para los movimientos individuales y para un movimiento combinado, a partir de una serie de aproximaciones lineales, como mínimos cuadrados, se pudieron determinar que los tres coeficientes con mejores resultados fueron:  $k_D = 0,5$ ,  $k_{D\theta} = 0,08$ ,  $k_\theta = 0,3$ .

Con el modelo cinemático ajustado para las condiciones de pruebas requeridas por el proyecto de investigación "Diseño e implementación de agentes robots, actuando de manera cooperativa, en una zona colapsada simulada, como apoyo a organismos de búsqueda y rescate urbano" y el modelo odométrico vectorial mostrado en la ecuación (6), se realizaron pruebas a la plataforma. Como zona de pruebas fue usado el coliseo de la Facultad Tecnológica de la Universidad Distrital Francisco José de Caldas en donde se realizó la simulación de una zona colapsada [3].

$$\begin{bmatrix} X(k+1) \\ Y(k+1) \\ \theta(k+1) \end{bmatrix} = \begin{bmatrix} x(k) + D(k) \cos\left(\theta(k) + \frac{\theta(k)}{2}\right) \\ y(k) + D(k) \sin\left(\theta(k) + \frac{\theta(k)}{2}\right) \\ \theta(k) + \frac{\theta(k)}{2} \end{bmatrix} + v(k) \quad (6)$$

Se implementó un algoritmo de búsqueda y evasión de obstáculos, mediante la utilización de sensores ultrasónicos y asimismo, mediante los modelos de cinemática y odometría se conocieron los desplazamientos de la plataforma y se establecieron rutas de navegación (figura 2).

Figura 2. Modelo de exploración de obstáculos a partir de la generación de trayectorias implementado en LabVIEW



Fuente: [3].

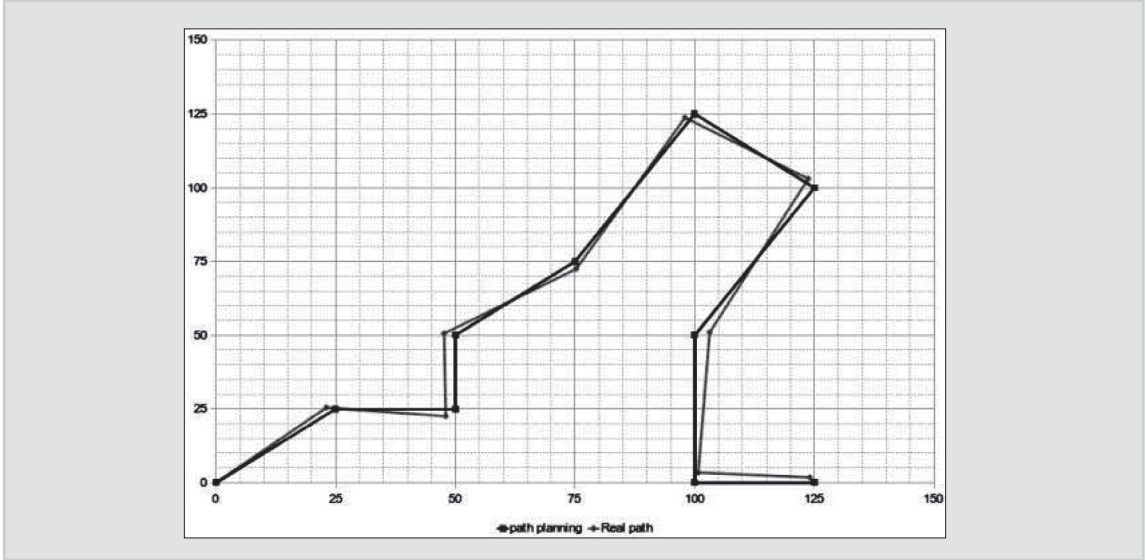
En la figura 3 se muestra el resultado de la implementación de una ruta predeterminada a la plataforma.

vido a los errores de odometría que aumentan constantemente y se van acumulando con cada desplazamiento.

Como se observa en la figura 3, la plataforma no sigue la trayectoria correctamente de-

Es aquí donde se implementa el filtro de Kalman, que estimará los parámetros correctos

Figura 3. Resultado obtenido para el seguimiento de una ruta predeterminada teniendo en cuenta los errores odométricos presentes en la plataforma móvil



44

Fuente: [3].

del robot tomando la información proveniente de los sensores.

### 3. Fases e implementación del filtro de Kalman

El filtro de Kalman calcula el estado del proceso en un instante y obtiene información (se realimenta) de la medida. Por tanto, las ecuaciones del filtro se pueden clasificar en dos tipos: actualización del tiempo (predicción) y actualización de las medidas (corrección) [2].

#### 3.1 Fase de predicción (t2)

En esta fase se calcula la estimación, es decir, la posición en que se encontrará la plataforma según el modelo cinemático y odométrico como se muestra en (7) [4]

$$X(k+1) = AX(k) + BU(k) + v(k) \quad (7)$$

Donde A y B son matrices dinámicas que se calculan como jacobianos del sistema:

Para A:

$$fx = X_{k+1} = X_k + \Delta D \cos\left(\theta + \frac{\Delta\theta}{2}\right)$$

$$fy = Y_{k+1} = Y_k + \Delta D \sin\left(\theta + \frac{\Delta\theta}{2}\right)$$

$$f\theta = \theta_{k+1} = \theta_k + \frac{\Delta\theta}{2}$$

$$A = \begin{bmatrix} 1 & 0 & -\Delta D \sin\left(\theta + \frac{\Delta\theta}{2}\right) \\ 0 & 1 & \Delta D \cos\left(\theta + \frac{\Delta\theta}{2}\right) \\ 0 & 0 & 1 \end{bmatrix}$$

Para B:

$$fx = X_{k+1} = X_k + (\Delta D + \varepsilon D) \cos\left(\theta + \left(\frac{\Delta\theta}{2} + \varepsilon\Delta\theta\right)\right)$$

$$fy = Y_{k+1} = Y_k + (\Delta D + \varepsilon D) \sin\left(\theta + \left(\frac{\Delta\theta}{2} + \varepsilon\Delta\theta\right)\right)$$

$$f\theta = \theta_{k+1} = \theta_k + \left(\frac{\Delta\theta}{2} + \varepsilon\Delta\theta\right)$$

Las anteriores son las ecuaciones del filtro de Kalman que se van a aplicar para calcular la posición de la plataforma móvil que incorpora los modelos de odometría y localización. El sistema de odometría nos da los incrementos en la posición del punto central del móvil respecto de unos ejes de referencia fijos, estos incrementos se tomarán como entradas directas al sistema. [5]

En la fase de estimación también se obtiene la matriz de covarianza del sistema, la cual almacena el error general de esta y está dada por la ecuación (8).

$$P(k+1) = A_k P_k A_k^T + B_k Q_k B_k^T \quad (8)$$

Donde  $\sigma^2$  es la varianza de las medidas tomadas y  $n$  es el número de medidas tomadas y el error característico de estado de la plataforma expresado en la ecuación (5)

#### 3.2 Fase de corrección

En esta parte del filtro se corrige la posición de la plataforma con la información que se recibe de los sensores; esta información está representada por la ecuación (9) [1].

$$z(k) = HX(k) + w(k) \quad (9)$$

Donde  $w(k)$  es el error asociado al sistema de medición y  $H$  es el jacobiano en función de la medida, y considerando que se mide directamente la posición y la orientación del robot se tiene:

$$H_\theta = \theta$$

$$H_y = y$$

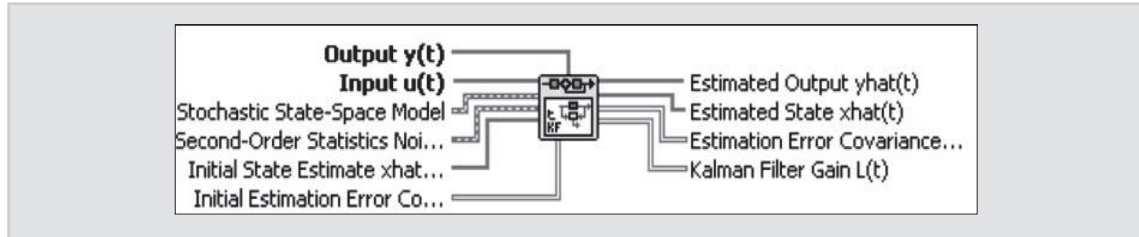
$$H_x = x$$

Por tanto

$$H = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

En la corrección también se incorporan las medidas tomadas por los sensores al estado anteriormente estimado así:

Figura 4. VI filtro de Kalman en LabVIEW



Fuente: elaboración propia.

$$X(k + 1) = X_k + K_k(Z_k - H(X_k, 0))$$

Posteriormente se actualiza la matriz de covarianzas, es decir se actualiza el error [5]:

$$P(k) = (I - K_k H_k) P_{(k+1)}$$

Y finalmente se obtiene la matriz de ganancias del filtro de Kalman, que contiene la nueva observación influenciada por el error entre la estimación y la medida. [6][7][8].

$$K(k) = P_{(k+1)} H_k^T (H_k P_k H_k^T + R_k)^{-1}$$

Donde  $R_k = E(w(k)w(k)^T)$

### 3.3 Implementación

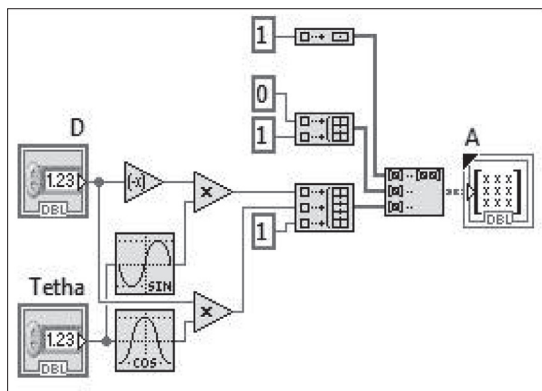
Para implementar el filtro de Kalman en el software LabVIEW de National Instruments, fue necesario adaptar las ecuaciones del siste-

ma a la herramienta; en la figura 4 se muestra el VI que calcula la estimación del filtro.

Donde  $y(t)$  es la entrada del sistema, es decir las mediciones de los sensores;  $u(t)$  es el estado inicial del sistema (para el cálculo de la matriz de covarianza no importa el estado inicial, puede ser cualquiera [5]); stochastic State-Space Model es el modelo del sistema; los jacobianos calculados, entradas y mediciones; Second-Order Statistics Noise es el modelo asociado al error del sistema y al error en las medidas;  $yhat(t)$  entrega el valor del estado estimado,  $xhat(t)$  entrega el modelo estimado, estimation error retorna el error de covarianza y  $L(t)$  es la matriz de ganancias del filtro [9].

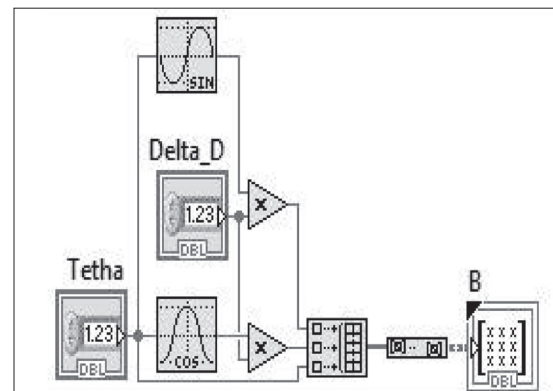
Una vez identificadas las entradas y salidas de filtro, se construyen las matrices que ingresan al sistema; A (figura 5), B (figura 6) y matrices (figura 7).

Figura 5. SubVI de la dinámica del sistema (Jacobiano A)



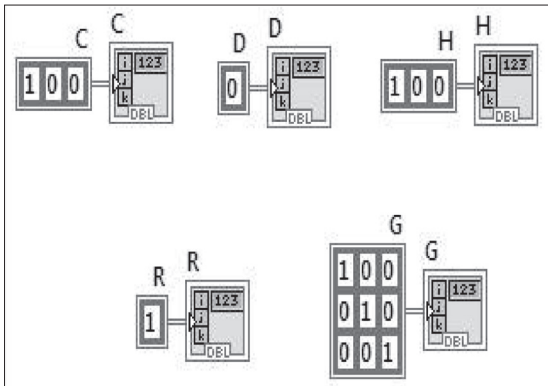
Fuente: elaboración propia.

Figura 6. SubVI de las entradas del sistema (x, y, theta)



Fuente: elaboración propia.

**Figura 7. SubVI C, D, H, R y G**



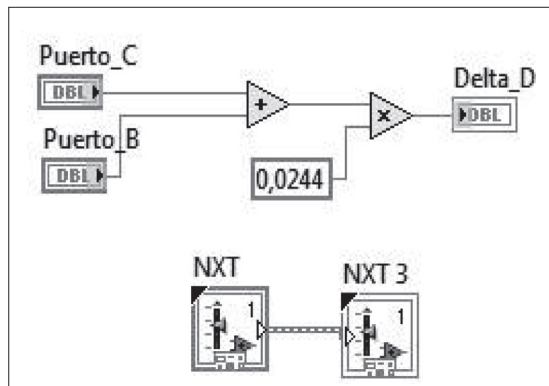
Fuente: elaboración propia.

Las entradas del sistema están dadas por  $X$ ,  $Y$  y  $\theta$ , los cuales dependen de la lectura de los sensores (encoders y compas); para esto se tienen dos SubVIs para la distancia ( $D$ ) y el ángulo ( $\theta$ ) (figura 8 y figura 9).

Después de construir el modelo del sistema y el modelo del error, se implementa el filtro de Kalman (figura 10).

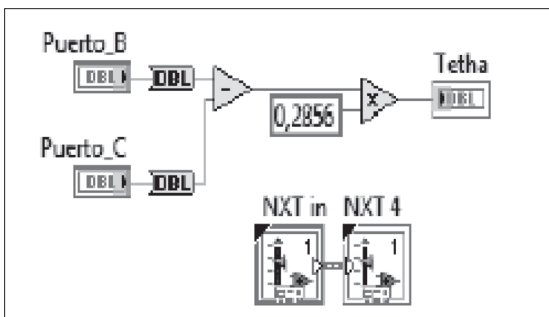
Para observar la respuesta del filtro se tomaron pruebas con trayectorias aleatorias; a continuación se muestran dos de ellas. En la primera (figura 11) la plataforma se detiene

**Figura 8. SubVI D**



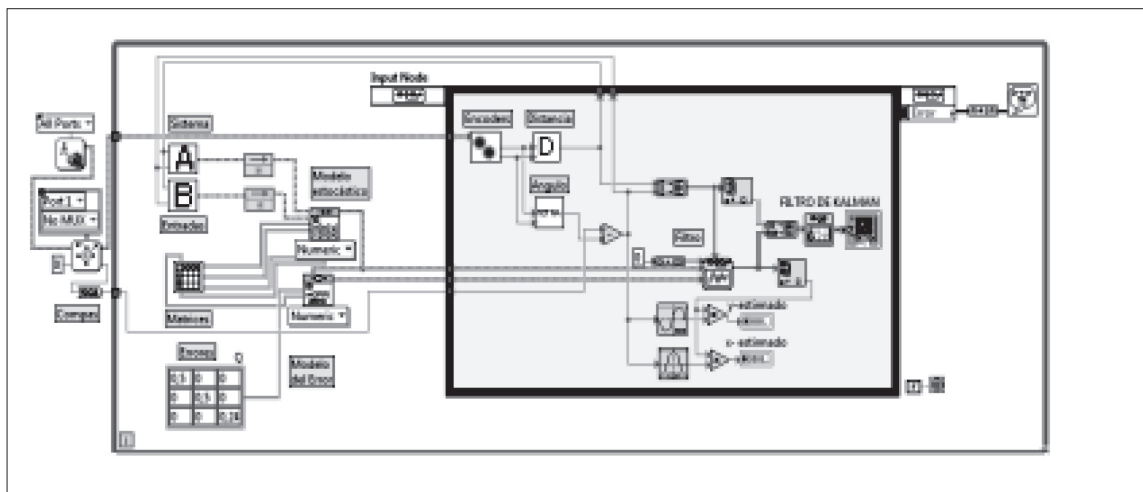
Fuente: elaboración propia.

**Figura 9. SubVI  $\theta$**



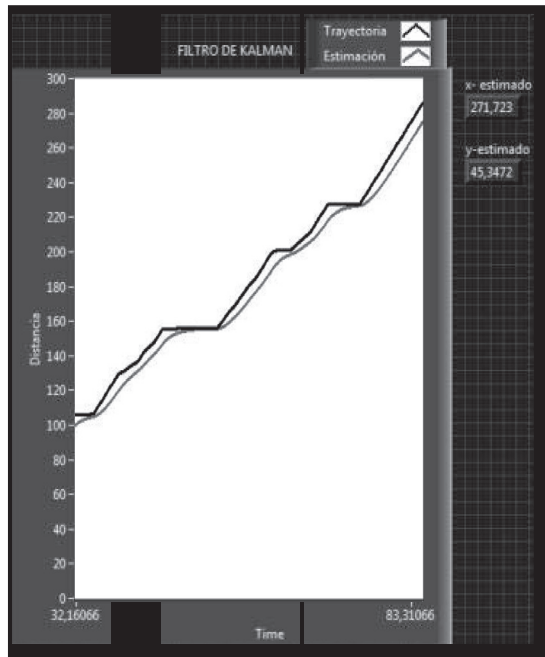
Fuente: elaboración propia.

**Figura 10. Implementación del filtro de Kalman**



Fuente: elaboración propia.

Figura 11. Respuestas del filtro de Kalman - prueba 1



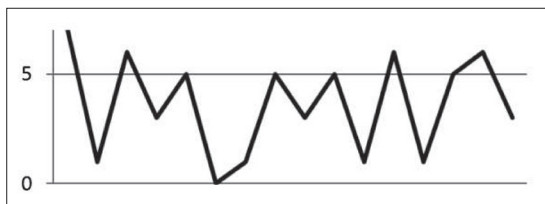
Fuente: elaboración propia.

en tres ocasiones; la trayectoria superior es la real y la inferior es la respuesta del filtro que resulta ser muy similar a la real, lo que indica que los valores de ruido son muy similares los reales.

En la figura 12 se obtiene el error en la estimación tomando los valores de las señales y restándolos.

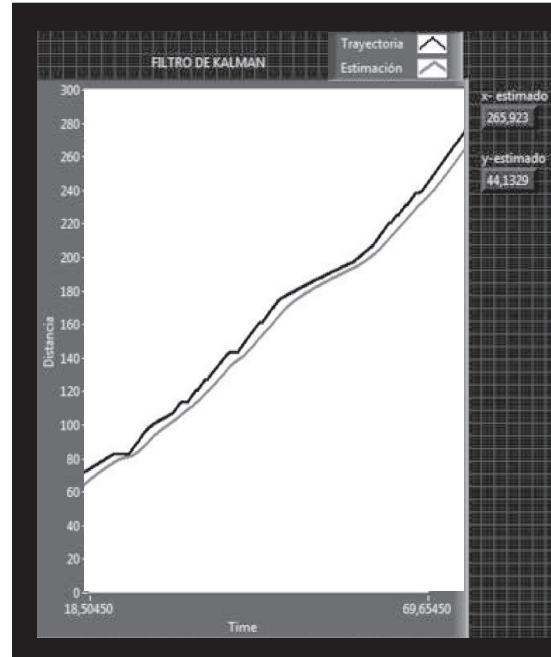
En la segunda prueba (figura 13) la plataforma no se detiene mientras navega por el entorno y de igual forma que en la prueba 1, la

Figura 12. Error en la estimación de la prueba 1



Fuente: elaboración propia.

Figura 13. Respuestas del filtro de Kalman - prueba 2



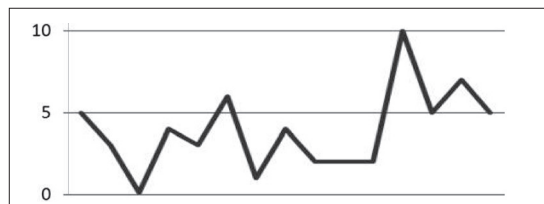
Fuente: elaboración propia.

respuesta del filtro (inferior) es muy similar a la trayectoria real del robot (superior). En la figura 14 se muestra el error de la estimación para esta prueba.

#### 4. Conclusiones

En el desarrollo del proyecto *Diseño e implementación de agentes robots, actuando de manera cooperativa, en una zona colapsada simulada, como apoyo a organismos de búsqueda y rescate urbano* fue necesario implementar el filtro de

Figura 14. Error en la estimación de la prueba 2



Fuente: elaboración propia.



Kalman para mejorar el proceso de localización del robot sobre la zona de exploración, ya que para algunas técnicas de rescate se siguieron rutas predefinidas y los errores de odometría podrían ocasionar lecturas incorrectas durante la búsqueda de la víctima; la técnica recursiva del filtro de Kalman permitió estimar la próxima posición de la plataforma correctamente con el ruido presente en el entorno.

Una vez corregidos los problemas de localización de cada plataforma; se realizará una comunicación ordenador robot con el objetivo de monitorear la posición del robot en instantes de tiempo determinados, lo cual permitirá dirigir a las plataformas a puntos específicos en el entorno; además se desarrollarán algoritmos de fusión sensorial implementados sobre LabVIEW y el mismo ladrillo NXT, que redundarán la información para hacer una estimación aún más exacta.

## 5. Agradecimientos

La autora agradece al Centro de Investigaciones y Desarrollo Científico de la Universidad distrital Francisco José de Caldas por la financiación del proyecto "Diseño e implementación de agentes robots, actuando de manera cooperativa, en una zona colapsada simulada, como apoyo a organismos de búsqueda y rescate urbano" y a la Facultad Tecnológica por el préstamo del espacio de prueba usado para el desarrollo del proyecto.

## 6. Referencias

- [1] M. Pinto, A. P. Moreira and A. Matos, *Localization of Mobile Robots Using an Extended Kalman Filter in a LEGO NXT*. IEEE Transactions on Education, 1-10. 2001.
- [2] P. R. Muñoz, T. Antonio and S. Montemayor, *DE IMÁGENES*. Autor.; 57. [En línea], disponible en <http://www.etsii.urjc.es/~asanz/documentos/MemoriaKalmanJun03.pdf>. 2003.
- [3] "Filtro de Kalman" Documentos Memoria de Kalman. Jun 2012. Retrieved from: <http://www.etsii.urjc.es/~asanz/documentos/MemoriaKalman.pdf>
- [4] R. G. Bermúdez, "Modelamiento cinemático y odométrico de robots móviles: aspectos matemáticos". Tecnura, vol. 20, no. 12. 2003.
- [5] F. Mart, P. Barrera, J. Mar, G. D. Rob, U. Rey y J. Carlos, *Localización basada en lógica difusa y filtros de Kalman para robots con patas*, 12. [En línea], disponible en <http://gsync.es/jmplaza/papers/comp2006-paco.pdf>. 2006.
- [6] P. Zarchan, *fundamentals of Kalman filtering: practical approach* volume 232 progress in astronautics and aeronautics, tercera edición 2009
- [7] G. Welch and G. Bishop, *An Introduction to the Kalman Filter*. In *Practice*, 1-16. 2003.
- [8] B. F. Aguado, J. A. Castaño, C. Parque, D. Boecillo, E. Z. Casanova, J. G. García-Bermejo y E. Paseo, *Diseño y simulación de un filtro kalman para un robot móvil*, 6. 2004.
- [9] Technology, I. *System Identification and Estimation in LabVIEW*. . [En línea], disponible en [http://home.hit.no/~hansha/documents/labview/training /System Identification and Estimation in LabVIEW/System Identification and Estimation in LabVIEW.pdf](http://home.hit.no/~hansha/documents/labview/training/System%20Identification%20and%20Estimation%20in%20LabVIEW.pdf). 2001.