

# Implementación del filtro de Kalman para la localización de un robot móvil tipo LEGO NXT en Labview

## *Implementation of the Kalman filter for localization of a mobile robot LEGO NXT in Labview*

**LEIDY YOLANDA LÓPEZ OSORIO**

Tecnóloga en Electrónica, Universidad Distrital Francisco José de Caldas. Estudiante de Ingeniería en Control, Universidad Distrital Francisco José de Caldas. Auxiliar de investigación adscrita al grupo de investigación en Robótica Móvil Autónoma (ROMA). Bogotá, Colombia. Contacto: *lylopezo@correo.udistrital.edu.co*

Fecha de recepción: 9 de Julio de 2012

Clasificación del artículo: Investigación

Fecha de aceptación: 1 de Octubre de 2012

Grupo de Investigación: ROMA

**Palabras clave:** *Odometría, Estimación, filtro de Kalman, LEGO NXT, LabVIEW.*

**Key words:** *Odometer, Estimation, Kalman filter, LEGO NXT, LabVIEW.*

### 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 diseñan diferentes rutas para la búsqueda de víctimas, las cuales deben ser seguidas por cuatro robots con la mayor exactitud posible. Durante el recorrido de las rutas se presentan errores de odometría, lo que dificulta la tarea del equipo de rescate. Este trabajo presenta la implementación del filtro de Kalman para estimar posiciones y poder corregir los errores en la localización de un robot.

### ABSTRACT

In the investigation project “Design and implementation of robots agents acting cooperatively, in a simulated collapsed area as support agencies urban search and rescue” different routes are designed to search for victims, which must be followed by four robots as accurately as possible, during the trip of the routes odometer errors occur, it difficult the task of the rescue team. This paper presents the implementation of the Kalman filter to estimate position and to correct errors in the location of a robot.

## 1. INTRODUCCIÓN

El análisis matemático de las características cinemáticas y odométricas de una plataforma móvil, permite obtener información a cerca de la posición de ésta, en donde los ángulos de giro y los desplazamientos son conocidos en todo instante.

Durante el desarrollo del proyecto de investigación se hizo dicho análisis, y para ejercer el control sobre la plataforma y mantenerla una ruta deseada durante la búsqueda de víctimas, se implementa el filtro de Kalman, que es una técnica recursiva 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 varianza 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 será posible corregir los errores y seguir las rutas correctamente.

El desarrollo de dicha técnica para la plataforma LEGO NXT se realiza en LabVIEW, ya que este software brinda una herramienta de estimación continua del Filtro de Kalman.

## 2. METODOLOGÍA Y RESULTADOS

La cinemática permite conocer la posición y orientación del móvil a partir de los parámetros físicos de la plataforma y determina expresiones matemáticas que permiten conocer el comportamiento del



**Figura 1.** Plataforma diferencial  
Fuente: LEGO Mindstorms

móvil a partir de la posición y orientación final. [3]

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  $Xm_L$  y  $\theta_L$  y son el desplazamiento y la orientación de la rueda izquierda,  $Xm_R$  y  $\theta_R$  y son el desplazamiento y la orientación de la rueda derecha y  $b$  la distancia entre las dos ruedas.

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

$$\Delta \theta = \frac{Xm_L - Xm_R}{b} = 0,2856(\theta_L - \theta_R) \quad (2)$$

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 es el vector de errores sistemáticos y  $v(k)$  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  $(t_k, t_{(k+1)})$  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  $v(k) \approx N(0, Q(k))$ , donde  $Q(k)$  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  $k_D$  el coeficiente de error de traslación de la plataforma relativo a  $\Delta D$  y expresado en  $[m^2/m]$ ;  $k_{D\theta}$  el coeficiente de error de rotación de la plataforma relativo a  $\Delta D$  y expresado en  $[rad^2/m]$ ; y  $k_\theta$  el coeficiente de error de rotación de la plataforma relativo a  $\Delta\theta$  y expresado en  $[rad^2/rad]$ .

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 Tec-

nológica de la Universidad Distrital Francisco José de Caldas en donde se realizó la simulación de una zona colapsada [3].

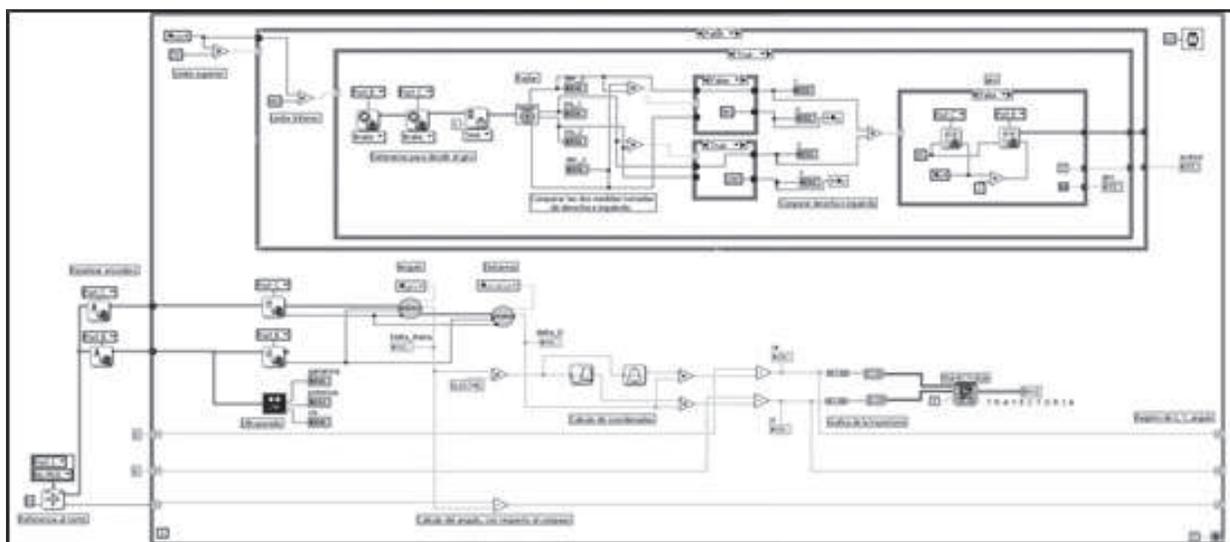
$$\begin{bmatrix} X(k+1) \\ Y(k+1) \\ \theta(k+1) \end{bmatrix} = \begin{bmatrix} x(k) + D(k)\cos\left(\theta(k) + \frac{\theta(k)}{2}\right) \\ y(k) + D(k)\sin\left(\theta(k) + \frac{\theta(k)}{2}\right) \\ \theta(k) + \frac{\theta(k)}{2} \end{bmatrix} + v(k) \quad (6)$$

Se implementó un algoritmo de búsqueda de obstáculos que permitiera evitar colisiones mediante la utilización de sensores ultrasónicos y asimismo, mediante los modelos de cinemática y odometría, se pudo establecer rutas de navegación para la plataforma móvil por su entorno (figura 2).

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

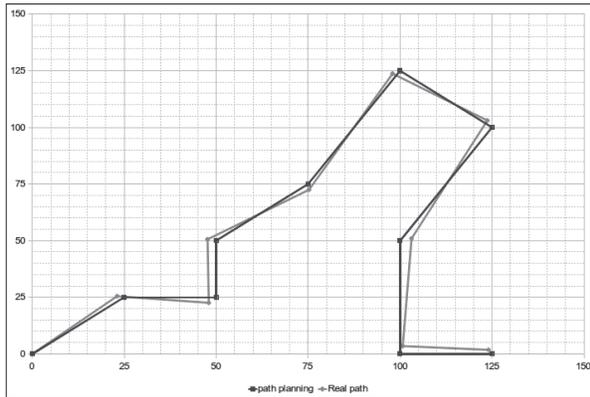
Como se observa en la figura 3, el robot no sigue la trayectoria correctamente; y es aquí donde se debe implementar el filtro de Kalman.

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].



**Figura 2.** Modelo de exploración de obstáculos a partir de la generación de trayectorias implementado en LabVIEW [3].

Fuente: Elaboración propia



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

Fuente: Elaboración propia

### Fase De Predicción:

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

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

Donde  $y$  se calculan como jacobianos de la función del sistema:

Para  $A$ :

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

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

$$f\theta = \theta_{(k+1)} = \theta_k + \Delta\theta/2$$

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

Para  $B$ :

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

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

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

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

Estas 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 del mismo 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  $P_K = \sigma^2/k$ , siendo  $\sigma$  es la varianza de las medidas tomadas y  $k$  es el número de medidas tomadas y  $Q_K$  el error característico de estado de la plataforma expresado en la ecuación (5)

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  $H$  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

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

$w(k)$ , es el error asociado al sistema de medición.

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

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

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

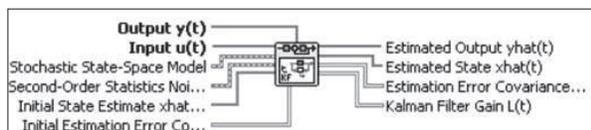
$$P(k) = (I - K_k H_k) P_{(k+1)}$$

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

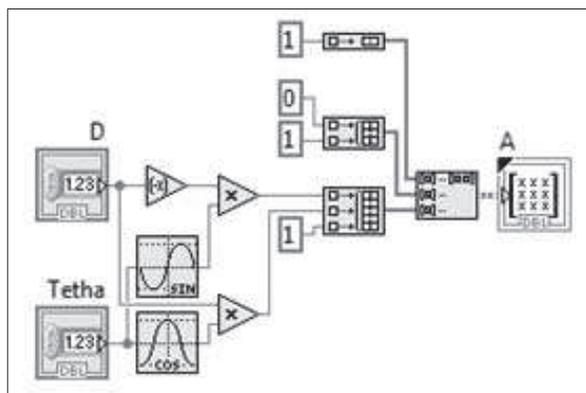
Para implementar el filtro de Kalman en LabVIEW fue necesario adaptar las ecuaciones del sistema a la herramienta; en la figura 4 se muestra el VI que calcula la estimación del filtro.



**Figura 4.** VI filtro de Kalman en Labview.  
Fuente: Elaboración propia

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 fil-



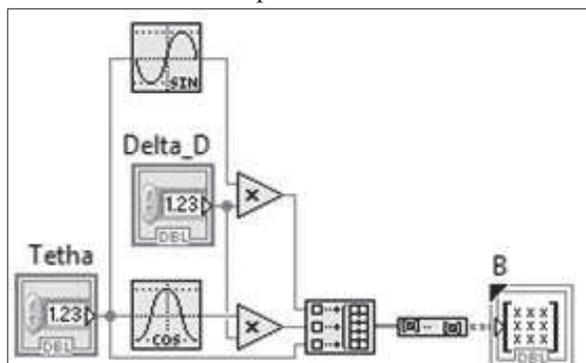
**Figura 5.** SubVI de la dinámica del sistema (Jacobiano A)  
Fuente: Elaboración propia

tro, se construyen las matrices que ingresan al sistema; A (figura 5), B (figura 6) y matrices (figura 7)

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).

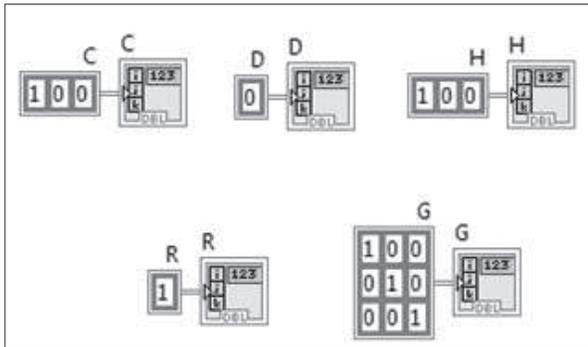
Después de construir el modelo del sistema y el modelo del error, se implementa el filtro de Kalman (figura 10).

Para observar la respuesta del filtro se tomaron

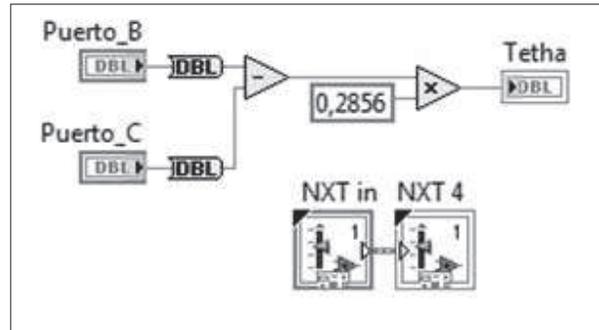


**Figura 6.** SubVI de las entradas del sistema (x, y, Theta)  
Fuente: Elaboración propia

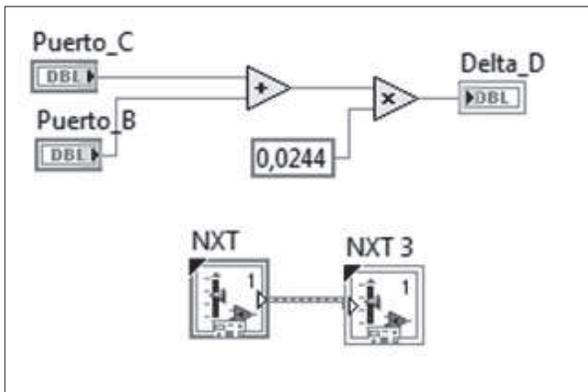
pruebas con trayectorias aleatorias; a continuación se muestran dos de ellas. En la primera (figura 11) la plataforma se detiene en tres ocasiones; la trayectoria superior es la real y la inferior es la res-



**Figura 7.** SubVI C, D, H, R y G  
Fuente: Elaboración propia



**Figura 9.** SubVI  $\Theta$   
Fuente: Elaboración propia

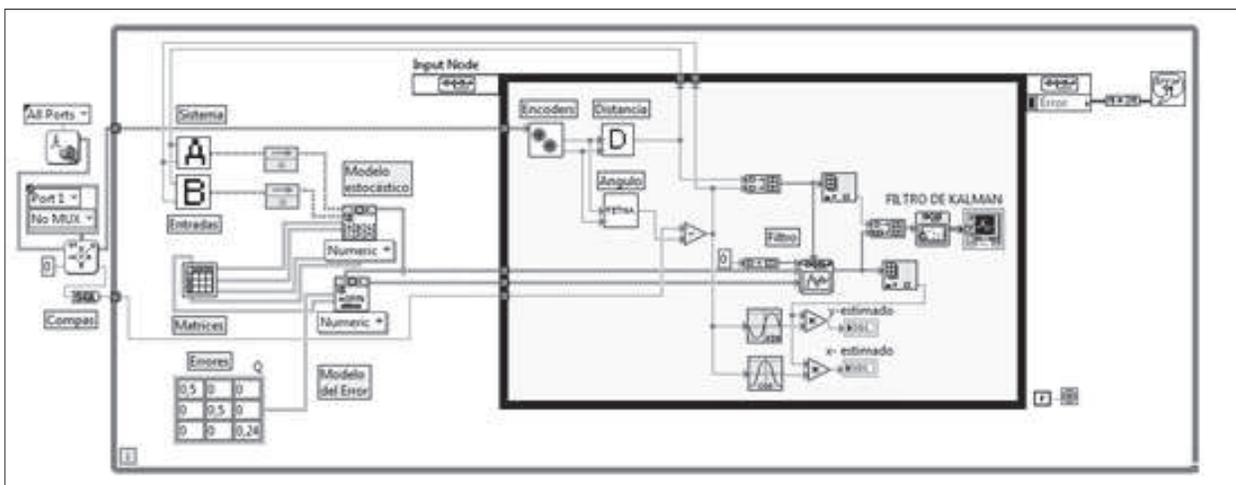


**Figura 8.** SubVI D  
Fuente: Elaboración propia

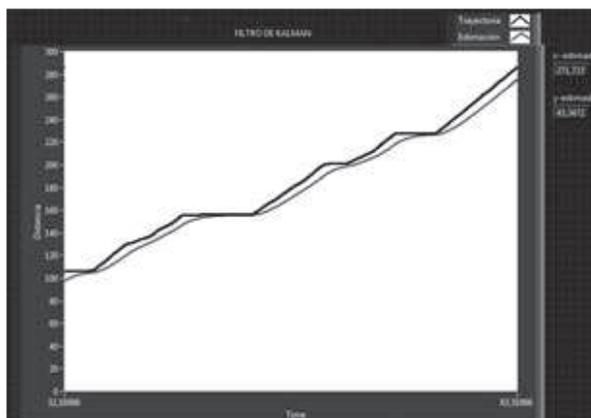
puesta del filtro que resulta ser muy similar a la real, lo que indica que los valores de ruido son muy parecidos a los verdaderos.

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 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



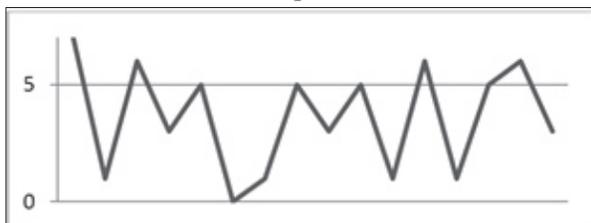
**Figura 10.** Implementación del filtro de Kalman.  
Fuente: Elaboración propia



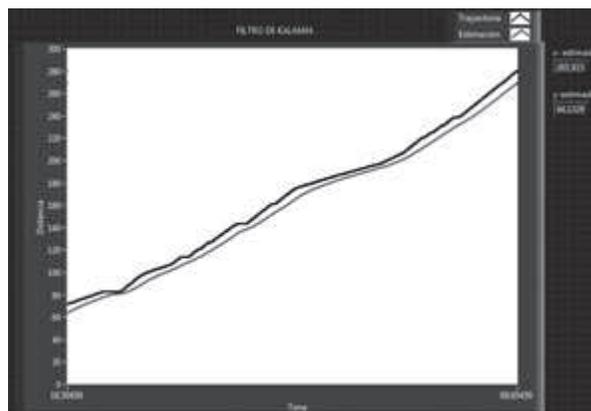
**Figura 11.** Respuestas del filtro de Kalman – prueba 1  
Fuente: Elaboración propia

### 3. 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 aplicar el filtro de Kalman



**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



**Figura 14.** Error en la estimación de la prueba 2  
Fuente: Elaboración propia

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 de éste 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.

### 4. FINANCIACIÓN Y 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.

---

**REFERENCIAS**

---

- [1]. M. Pinto, A. P. Moreira, & A. Matos, *Localization of Mobile Robots Using an Extended Kalman Filter in a LEGO NXT*. IEEE Transactions on Education, pp 1-10, 2011.
- [2]. M. Pinto and G. Bermúdez, “Determinación de parámetros de un robot móvil de Lego Mindstorms,” *Ingeniería, Investigación y Desarrollo*, vol. 5, no. 2, pp. 7-13, 2007.
- [3]. P. Muñoz, “*Aplicación del filtro de Kalman al seguimiento de objetos en secuencias de imágenes*,” tesis, Universidad Rey Juan Carlos, España, 2003.
- [4]. R G. Bermúdez, “*Moldeamiento cinemático y odométrico de robots móviles: aspectos matemáticos*,” *Tecnura*, vol. 12, no. 20, 2003.
- [5]. F. Mart, P. Barrera, J. Mar, G. D. Rob, U. Rey, & Carlos, J. (n.d.). *Localización Basada en Lógica Difusa y Filtros de Kalman para Robots con Patas* 2006.
- [6]. P. Zarchan, “Fundamentals of Kalman filtering: practical approach” volume 232 progress in astronautics and aeronautics, tercera edición 2009
- [7]. G. Welch, & G. Bishop. “An Introduction to the Kalman Filter”. In Practice, pp 1-16, 2003.
- [8]. B. F. Aguado, J. A. Castaño, C. Parque, D. Boecillo, E. Z. Casanova, J. G. García-bermejo, & E. Paseo, *Diseño Y Simulación De Un Filtro Kalman Para Un Robot Móvil*, 6, 2004.
- [9]. Technology, I. System Identification and Estimation in LabVIEW. 2011, En línea Disponible:[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/System%20Identification%20and%20Estimation%20in%20LabVIEW.pdf)