Reducción del efecto del retardo en el control de un robot móvil diferencial

H. Pérez-Villeda * **F. Mirelez-Delgado** *,** **A. Morales-Díaz** *

* *Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional Unidad Saltillo*
Ramos Arizpe, Coah. 25900, México
Teléfono: (844) 4389600 Ext. 8500
** *flabio.mirelez@cinvestav.edu.mx*

Resumen: En este artículo el algoritmo de filtro de partículas se utiliza para compensar los efectos negativos del retardo generado por los sistemas de comunicación en el seguimiento de trayectorias de un robot móvil diferencial. Este método demostró atenuar los efectos originados por el retardo temporal en el envío y recepción de los datos. Se presentan los resultados experimentales obtenidos y un análisis detallado del funcionamiento algoritmo con diferentes parámetros de entrada.

Palabras clave: Filtro de partículas, robot móvil diferencial, Retardo.

1. INTRODUCCIÓN

Los sistemas de comunicación usados en robótica presentan retardos inherentes tanto en la emisión como en la recepción de datos, lo que puede afectar el desempeño de los robots en sus tareas e incluso inducir inestabilidad en el control.

En el caso particular de los robots móviles, éstos se utilizan para exploración, evasión de obstáculos, formación, trasporte de objetos, entre otras aplicaciones. Para obtener un desempeño efectivo de estos sistemas en este tipo de tareas, es necesario que el error de seguimiento de trayectoria sea mínimo, sin embargo un retardo en los sistemas de comunicación puede causar que la trayectoria no sea seguida de manera efectiva. Lo anterior se debe a que el cálculo del control se realiza con un estado retardado del robot, implicando que el control calculado en ese instante de tiempo se aplique cuando el robot se encuentra en otra posición diferente. Algunas investigaciones se han enfocado en resolver este problema con técnicas diferentes, por ejemplo en Kawai et al. (2010) los retardos de comunicación se consideran para realizar teleoperación con un robot virtual. Por otra parte en Sira-Ramírez et al. (2010) se muestra un control basado en observador para compensar el retardo y conseguir el seguimiento de trayectorias en un robot omnidireccional. En Kim et al. (2011) se propone un algoritmo para evasión de obstáculos que considera el retardo en un grupo de robots móviles, a diferencia de este trabajo, la corrección en la localización de cada robot se realiza con un filtro de Kalman.

En este trabajo el método propuesto para disminuir los efectos del retardo es el algoritmo de filtro de partículas (FP). El filtro de partículas es un método probabilístico que se utiliza en este trabajo para predecir el estado futuro del robot a partir del estado retardado obtenido por el sensor (una cámara en cielo). Algunas de las aplicaciones más recientes del filtro de partículas en robótica se pueden encontrar en Duan et al. (2007); Yu et al. (2008); Yin et al. (2010); Hsu et al. (2010); Wardhana et al. (2013), donde su uso se centra en la localización del robot móvil. Por otra parte en Rusdinar et al. (2010) esta técnica también es usada para realizar mapeo y localización simultánea (SLAM).

El artículo está estructurado de la siguiente manera: En la sección 2 se presenta el modelo cinemático de un robot móvil y la estructura del control usado en este trabajo. En la sección 3 se describen los componentes necesarios para la realización de los experimentos. La sección 4 plantea la problemática que surge cuando un sistema robótico se ve afectado por los retardos en la comunicación. La metodología a seguir para resolver el problema planteado se muestra en la sección 5. Para la validación de la solución propuesta se realizaron varias pruebas de las cuales las más significativas se muestran en la sección 6. Por último la sección 7 presenta las conclusiones y el trabajo a futuro.

2. CONTROL DE UN ROBOT MÓVIL

El modelo cinemático de un robot móvil está dado por las siguientes ecuaciones:

$$\begin{aligned} \dot{x} &= v\cos(\theta) \\ \dot{y} &= v\sin(\theta) \\ \dot{\theta} &= \omega \end{aligned} \quad (1)$$

Donde v es la velocidad lineal del robot móvil y ω la velocidad angular, además se considera la restricción no-holónoma $\dot{x}\sin(\theta) - \dot{y}\cos(\theta) = 0$.

Una representación en el plano cartesiano de un robot móvil con centro de masa en medio del eje de las llantas y con una rueda pasiva puede apreciase en la figura 1, donde ξ son las coordenadas generalizadas Cartesianas, $(x,\ y)$ es la posición del centro de masa del robot en el plano y θ denota el ángulo de orientación.

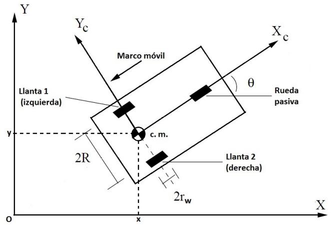


Fig. 1. Estructura del retardo en el sistema de comunicación.

Para el seguimiento de trayectorias para este tipo de robots existen diferentes controles que se pueden encontrar en la literatura actual, sin embargo, en este trabajo se ha usado un control no lineal propuesto en A. Morales (2013) , debido a los buenos resultados que este control ha reportado. Dicho control consiste en el cálculo de una velocidad lineal y una velocidad angular y se muestra a continuación.

$$
\begin{aligned}
v_i &= v_{ri} cos\theta_{ei} + \\
&\quad C_{ix}\left[x_{ei} + \sum_{j=1, j\neq i}^{n} C_{ijx}(x_{ei} - x_{ej})\right] \\
\omega_i &= \omega_{ri} + C_{i\theta}\theta_{ei} + vr_i \frac{sin\theta_{ei}}{\theta_{ei}} \frac{K}{\alpha_i} C_{iy} \\
&\quad \left[y_{ei} + \sum_{j=1, j\neq i}^{n} C_{ijy}(y_{ei} - y_{ej})\right]
\end{aligned} \tag{2}
$$

Donde C_{ix}, C_{ijx}, C_{iy}, C_{ijy}, $C_{i\theta}$ y K son las ganancias del control y el término α_i delimita el efecto de los errores de seguimiento y acoplamiento, definiéndose como:

$$
\alpha_i = \sqrt{K^2 + x_{ei}^2 + y_{ei}^2 + \beta_{ij}} \tag{3}
$$

$$
\beta_{ij} = \sum_{j=1, j\neq i}^{n} (x_{ei} - x_{ej})^2 + \sum_{j=1, j\neq i}^{n} (y_{ei} - y_{ej})^2 \tag{4}
$$

De acuerdo con el plano de referencia y considerando la postura deseada (véase la Figura 2), el error de seguimiento se define como:

$$
\begin{aligned}
x_i^e &= (x_i^d - x_i)\cos\theta_i + (y_i^d - y_i)\sin\theta_i \\
y_i^e &= -(x_i^d - x_i)\sin\theta_i + (y_i^d - y_i)\cos\theta_i \\
\theta_i^e &= \theta_i^d - \theta_i
\end{aligned} \tag{5}
$$

La representación de este control en este trabajo será una función $C(v(t), \omega(t))$ conformado por una velocidad lineal y una velocidad angular respectivamente.

3. PLATAFORMA EXPERIMENTAL

La plataforma experimental consiste en una cámara en cielo y un robot móvil (ver Fig. 3). La cámara es utilizada como sensor para obtener posición y orientación del robot móvil mediante procesamiento de imágenes. La cámara trabaja a 30 cuadros por segundo (fps). El área de trabajo cubierta con esta cámara es de 2.1 x 1.5 [m]. El robot móvil es un robot diferencial de la marca

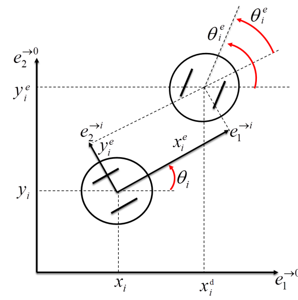


Fig. 2. Posición actual y deseada de un robot móvil diferencial.

iRobot Create®. La comunicación entre el robot y la computadora es mediante el protocolo bluetooth.

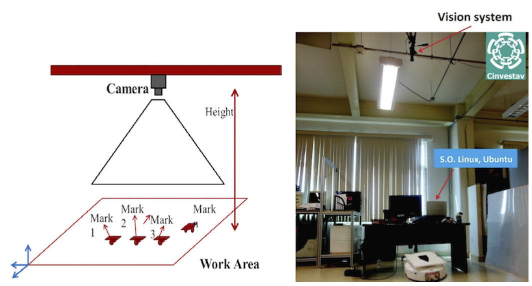


Fig. 3. Plataforma experimental.

En la Figura 4 se muestra la estructura del retardo en el sistema de comunicación. Se considera un retardo en la recepción del estado del robot, que se obtiene por el sensor (cámara) y otro retardo en el envío de la acción del control calculado.

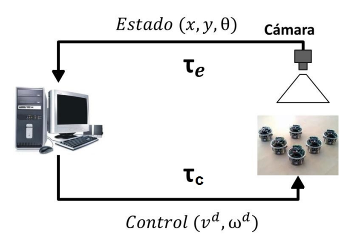


Fig. 4. Estructura del retardo en el sistema de comunicación.

El estado del robot tarda un lapso de tiempo en llegar a la computadora generando así un retardo en el envío del estado que nombraremos τ_e, mientras que el control calculado tarda otro lapso de tiempo desde que se envía desde la computadora hasta que llega al robot generando otro retardo que llamaremos τ_c.

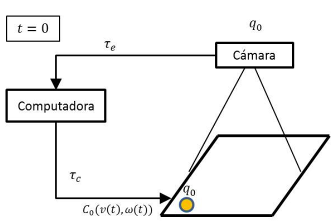
4. LÓGICA DE FUNCIONAMIENTO DEL ROBOT MÓVIL CON RETARDO

Considerando el esquema de retardo de la Figura 4, la evolución del funcionamiento del robot trabajando bajo retardo se muestra en la figura 5 y se describe a continuación. En la Figura 5a se observa al robot en un estado inicial q_0, el cual es capturado por la cámara. En ese mismo instante de tiempo una acción control $C_0(v(t), \omega(t))$ ha llegado al robot.

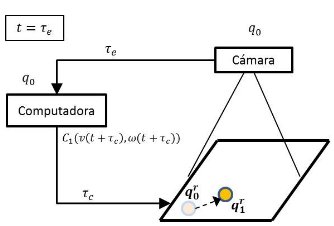
En la Figura 5b se observa que ya ha transcurrido un lapso de tiempo τ_e, es decir, el tiempo que tardó en llegar el estado censado por la cámara hacia la computadora, sin embargo, el robot se mantuvo en movimiento durante este lapso de tiempo debido a la acción de control inicial $C_0(v(t), \omega(t))$ ejecutada el instante t_0, así que en el instante de tiempo $t = \tau_e$ el robot se encuentra ya en un estado real q_1. No obstante, la computadora solo conoce el estado retardado q_0. En este instante de tiempo se debe calcular la nueva acción de control $C_1(v(t), \omega(t))$ pero no para el estado q_1 ya que se considera que el control calculado no llegará instantáneamente sino tras un lapso de tiempo τ_c así que el control se debe calcular en función de una predicción del estado. Esta predicción realizada se basa en calcular el estado en el cual se encontrará el robot tras el control aplicado $C_0(v(t), \omega(t))$ durante un lapso de tiempo $\Delta t = [(t - \tau_e)\ \ (t + \tau_c)]$ a partir del estado inicial q_0, así que el control que se calcula en este instante de tiempo es $C_0(v(t + \tau_c), \omega(t + \tau_c))$.

Finalmente, en la Figura 5c, se observa que la acción de control $C_1(v(t), \omega(t))$ ha llegado al robot tras transcurrir un tiempo τ_c. El control se aplicará al robot en un estado q_2, el cual fue obtenido por la predicción de la Figura 5b y con base en dicho estado fue como se calculó $C_1(v(t), \omega(t))$.

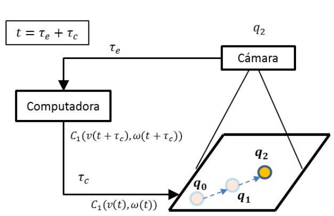
Posteriormente el estado q_2 pasa a convertirse en el estado q_0 ($q_0 = q_2$) aplicándose la misma metodología de forma recursiva para los siguientes estados.



(a)



(b)



(c)

Fig. 5. Lógica de funcionamiento.

5. METODOLOGÍA

La metodología de este trabajo puede dividirse en 3 etapas principales:

1. Estimación previa basada en el modelo cinemático.
2. Estimación final basada en el Filtro de partículas.
3. Calculo de la nueva acción de control.

El objetivo de usar el filtro de partículas es poder estimar el estado futuro del robot para que cuando reciba una acción de control ésta corresponda con el estado futuro real del móvil. Esta estimación se hace a partir de una acción de control retardada aplicada al robot cuando se tomó la lectura del estado del móvil. La magnitud total del retardo también es considerada para la estimación del estado del robot, así como el modelo cinemático del robot diferencial.

El modelo del robot es utilizado para tener una estimación del estado $[x(t)\ y(t)\ \theta(t)]$ dada una acción de control y un estado inicial. Posteriormente, el algoritmo del filtro de partículas se encarga de realizar una nueva estimación considerando un conjunto de partículas obtenidas en una iteración anterior. Esta estimación realizada $[x(t + \tau_c)\ y(t + \tau_c)\ \theta(t + \tau_c)]$ por el filtro de partículas es el estado de entrada con el que se calcula la acción de control(ec. 2) para el robot en su estado actual. A continuación se explican a detalle las tres etapas del método implementado.

Etapa 1

La primera estimación basada en el modelo cinemático del robot dado por:

$$\dot{X} = \begin{bmatrix} \dot{x}_i \\ \dot{y}_i \\ \dot{\theta}_i \end{bmatrix} = \begin{bmatrix} v_i \cos(\theta_i) \\ v_i \sin(\theta_i) \\ \omega_i \end{bmatrix} \tag{6}$$

Como resultado se obtiene una primera predicción del estado en el cual se encontrará el robot cuando reciba la nueva acción de control. Esta estimación hace uso del estado retardado del sistema $q(t - \tau) \in \mathbb{R}^3$, de la última acción de control C_{k-1}, y del retardo total τ_T donde:

$$q(t-\tau) = \begin{bmatrix} x(t - \tau_e) \\ y(t - \tau_e) \\ \theta(t - \tau_e) \end{bmatrix} \quad C_{k-1} = \begin{bmatrix} v(t - \tau_e) \\ \omega(t - \tau_e) \end{bmatrix} \quad \tau_T = \tau_e + \tau_c \tag{7}$$

La predicción de la etapa 1 consta de una integración del modelo cinemático del robot móvil durante un intervalo $\Delta t = [(t - \tau_e)\ (t + \tau_c)]$, tomando como condición inicial el estado retardado $q(t - \tau)$ y la última acción de control C_{k-1}. Con esta integración se obtiene la primera estimación q_1^e. La expresión matemática de esta primera estimación sería:

$$q_1^e = \int_{t-\tau_e}^{t+\tau_c} \dot{X}(t)dt \tag{8}$$

La razón de la integración del modelo cinemático durante el intervalo mencionado Δt se ha justificado en la descripción de la Figura 5b. Para realizar la primera estimación de un determinado estado q_τ con una acción de control

C_{k-1} durante un intervalo de tiempo Δt se ha creado una función llamada *estimador*.

$$q_1^e = estimador(\Delta t, q_\tau, C_{k-1}) = \int_{t-\tau_e}^{t+\tau_c} \dot{X}(t)dt \qquad (9)$$

Etapa 2

Esta etapa consiste en hacer una nueva estimación utilizando el filtro de partículas. Este algoritmo se describe con detalle en la tabla 1. El algoritmo necesita como argumentos de entrada el conjunto de partículas obtenido por en la iteración anterior (P_{t-1}), la observación del sistema, que en este caso será la estimación q_1^e obtenida en la etapa 1, la acción de control anterior C_{k-1} , el intervalo de tiempo $\Delta t = [(t-\tau_e)\ \ (t+\tau_c)]$ y finalmente una desviación estándar σ.

Cabe mencionar que para hacer la estimación de cada partícula del conjunto P_{t-1}, con una acción de control C_{k-1} durante un intervalo de tiempo Δt es necesario hacer uso de la función *estimador* de la etapa 1 ya que para cada partícula obtenida del filtro de partículas se obtiene una estimación futura. Como argumentos de salida se obtiene una nueva estimación del estado del robot q_1^{e*} y un nuevo conjunto de partículas P_f que se utilizarán en la siguiente iteración.

Tabla 1. Algoritmo: filtro de partículas (P_{t-1})

```
P̄_t = P_t
for i = 1 hasta N hacer
    Muestrear q_t^i ~ p(q_t | q_{t-1}^i)
    Asignar peso de partícula w_t^i = p(z_t | q_t^i)
end for
Calcular peso total k = Σ_{t=1}^{N} w_t^i
for i = 1 hasta N hacer:
    Normalizar w_t^i = k^{-1} w_t
    P̄_t = P̄_t + {q_t^i, w_t^i}
end for
(q_1^{e*}, P_t) = Remuestreo(P̄_t)
Regresar (q_1^{e*}, P_t)
```

- $\bar{P}t, Pt$ es el conjunto de partículas
- N es el número de partículas
- q_t^i es el estado de una partícula
- (q_1^{e*}, P_t) nueva estimación del estado del robot
- w_t es el peso de una partícula
- $p(q|q_t^i)$ distribución de transición
- $p(z|q_t^i)$ distribución de medición

Etapa 3

Una vez obtenida la estimación del estado q_1^{e*} se calcula la nueva señal de control C_{k-1} mediante el uso de las ecuaciones de la expresión (2). Las tres etapas antes mencionadas se resumen en la Tabla 2.

6. EXPERIMENTOS Y RESULTADOS

Para probar la funcionalidad del método propuesto se realizaron diferentes pruebas experimentales donde el objetivo es que el robot móvil siga una trayectoria deseada a pesar de variaciones en la magnitud del retardo. Se

Tabla 2. Algoritmo: Estimación del estado basada en filtro de partículas.

```
Mientras (t ≤ tiempo de simulación)
    Etapa 1: Estimar q_{f1}
        C_{k-1} = C_k
        q_1^e = estimador(Δt, q_τ, C_{k-1}) = ∫_{t-τ_e}^{t+τ_c} Ẋ(t)dt
    Etapa 2: Estimar q_1^{e*}
        P_{t-1} = P_f
        [q_1^{e*}, P_f] = algoritmoF.P.(Δt, C_{k-1}, q_1^e, P_{t-1}, σ)
    Etapa 3: Calcular el control C_k
        C_k = control(t + τ_c, q_1^{e*})
Terminar
```

considera que el retardo de envío y de recepción son iguales $\tau_e = \tau_c$, la trayectoria deseada son 5 vueltas en un círculo con radio de 0.5 [m] centrado en el origen del plano Cartesiano de referencia, además se establece una restricción de tiempo no mayor a 80 [seg] para completar el recorrido.

Prueba 1

Los parámetros para esta prueba se muestran en la Tabla 3 y los resultados se presentan en las Figuras 6 y 7.

Tabla 3. Parámetros usados para la prueba 1.

Parámetro	**Valor**
Retardo $\tau_e = \tau_c$	100 [mseg]
Condición inicial deseada	$q_0^d = [0{,}5,\ 0,\ 1{,}5708]^T$
Condición inicial real sin FP	$q_0^r = [0{,}5125,\ -0{,}008,\ 1{,}5541]^T$
Magnitud del error inicial sin FP	0.0223
Condición inicial real con FP	$q_0^r = [0{,}5165,\ -0{,}008,\ 1{,}5871]^T$
Magnitud del error inicial con FP	0.0245
Número de partículas	1000
Desviación estándar	$\sigma_x = \sigma_y = \sigma_\theta = 1$

De acuerdo con los resultados experimentales obtenidos se puede observar que los efectos negativos del retardo comienzan a partir de $\tau_e = \tau_c = 100\ [ms]$ aproximadamente. Se puede ver que la condición inicial real del robot está muy cerca de la condición inicial deseada. El comportamiento natural del robot con retardo comienza a presentar oscilaciones crecientes en los errores de y y θ provocando que la trayectoria del robot se deforme de manera incremental y no forme un círculo.

Al aplicar el método propuesto se observa en la figura 7 que las oscilaciones de los errores se atenúan. Por otra parte, en las gráficas de estimación se observa que la predicción realizada por el filtro de partículas con respecto al estado futuro es acertada, puesto que la gráfica del estado real coincide con la gráfica del estado estimado.

Prueba 2

De manera análoga a la prueba anterior, los parámetros para esta prueba se muestran en la Tabla 4 y los resultados en las Figuras 8 y 9.

En esta prueba, el retardo tiene un valor de 150 $[ms]$, lo cual provoca que el sistema sea inestable. En las

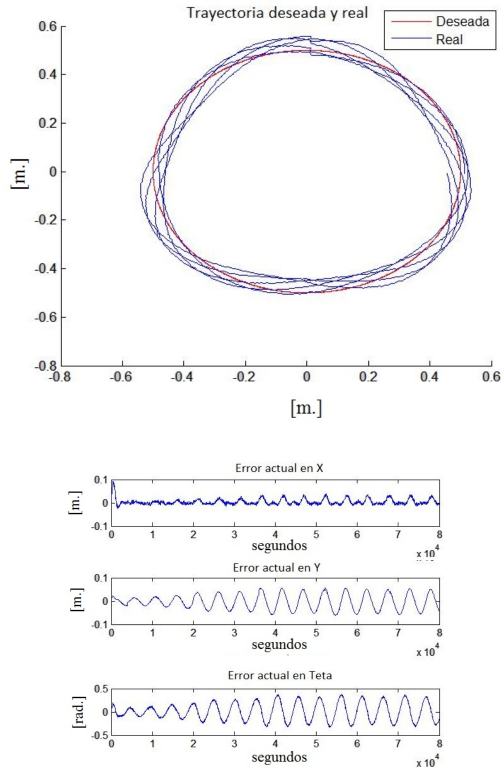


Fig. 6. Resultados de seguimiento Prueba 1 sin filtro de partículas.

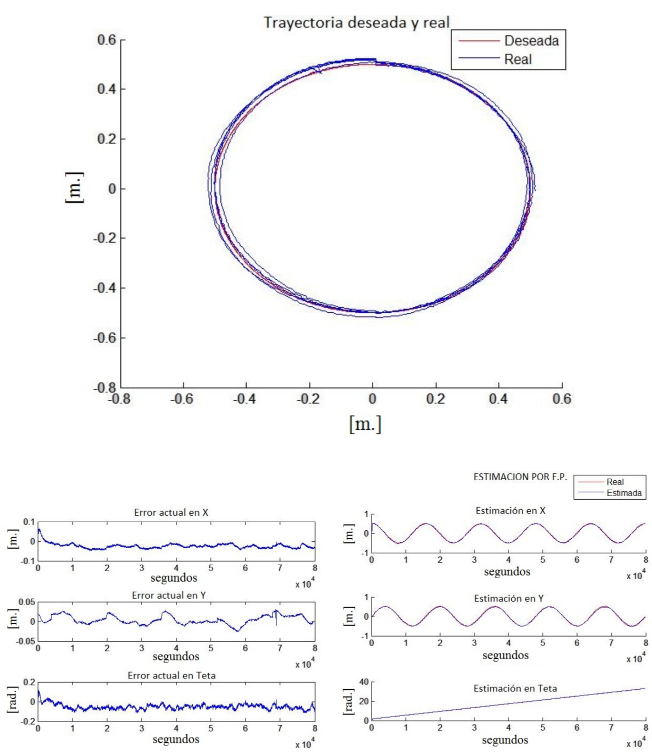


Fig. 7. Resultados de seguimiento Prueba 1 usando filtro de partículas.

gráficas de los errores de trayectoria de la Figura 8, se observan nuevamente oscilaciones con grandes amplitudes. Este comportamiento se ve reflejado en la gráfica

Tabla 4. Parámetros usados para la prueba 2.

Parámetro	**Valor**
Retardo $\tau_e = \tau_c$	150 [ms.]
Condición inicial deseada	$q_0^d = [0{,}5m.\ 0m.\ 1{,}5708rad.]^T$
Condición inicial real sin FP	$q_0^r = [0{,}508m.\ -0{,}0241m.\ 1{,}5213rad.]^T$
Magnitud del error inicial sin FP	0.0339
Condición inicial real con FP	$q_0^r = [0{,}5296m.\ 0{,}001m.\ 1{,}6457rad.]^T$
Error de condición inicial con FP	0.0805 m.
Número de partículas	10000
Desviación estándar	$\sigma_x = \sigma_y = \sigma_\theta = 2$

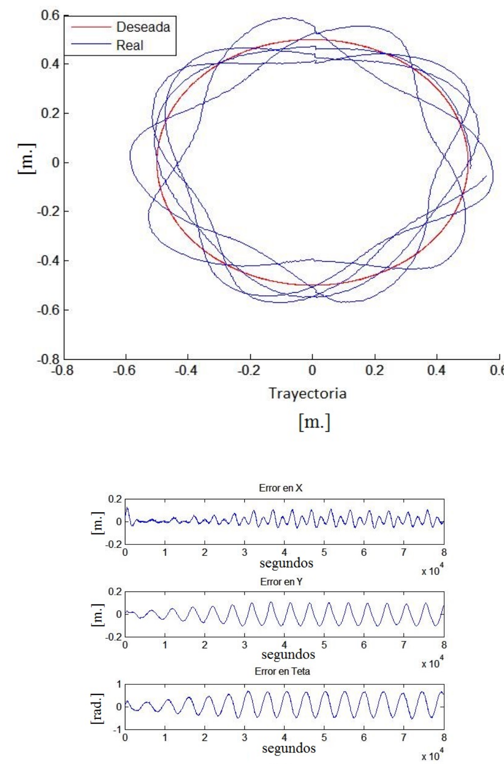


Fig. 8. Resultados de seguimiento Prueba 2 sin filtro de partículas.

de trayectorias donde se puede observar que la ruta real del robot se ha deformado completamente. Esta prueba se realizó tomando en cuenta un estado inicial real del robot cercano al estado inicial deseado. Tras aplicar el filtro de partículas con los parámetros especificados en la Tabla 4, en las gráficas de estimación por el filtro de partículas (figura 9) se observa que prácticamente las gráficas de la trayectoria real están sobre puestas con las gráficas de estimación para cada trayectoria x, y, θ, lo que quiere decir que la estimación q_1^{e*} es bastante buena. En la gráfica de trayectoria de esta misma figura 9 se observa que el comportamiento del robot mejora y las oscilaciones de los errores en x y y se atenúan significativamente.

7. CONCLUSIONES Y TRABAJO A FUTURO

Tras analizar los resultados experimentales se han formulado las siguientes conclusiones.

1. Las condiciones iniciales reales del robot deben de encontrarse muy cercanas a las condiciones iniciales deseadas. La razón es porque entre más cerca se encuentre el robot de su estado inicial deseado, el

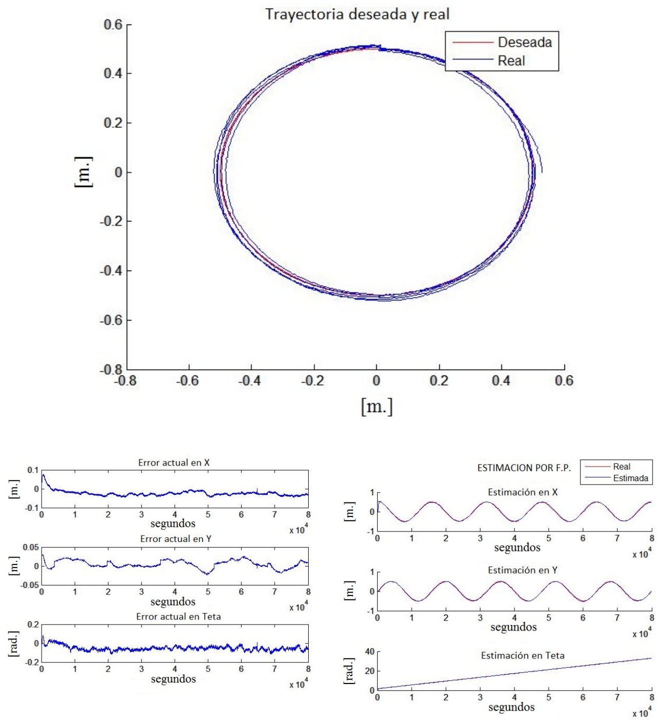


Fig. 9. Resultados de seguimiento Prueba 2 con filtro de partículas.

error es más pequeño, por lo tanto la velocidad de respuesta requerida del robot no es muy grande, lo que hace más exacta la estimación del estado futuro, considerando que la etapa 1 y la etapa 2 del método propuesto en este trabajo consisten respectivamente de una integración durante un lapso de tiempo $\Delta t = [(t - \tau_e)\ \ (t + \tau_c)]$ y un conjunto de partículas P_f.

2. Entre más grande sea el error de estado inicial es necesario aumentar la desviación estándar, sin embargo, para errores del estado inicial mayores a 0.15 el algoritmo tiene dificultades para converger.
3. El algoritmo funciona de manera eficiente para retardos menores a 200 $[ms]$.
4. Si el retardo crece es necesario aumentar la desviación estándar para tener mejores resultados.
5. Aumentar el número de partículas en teoría proporciona mejores resultados pero implica mayor número de cálculos, lo que implica aumento del retardo.
6. La convergencia del algoritmo también depende de la velocidad del robot móvil, bajo este esquema, el algoritmo convergerá para tiempos mayores puesto que el robot tendría más tiempo para completar la trayectoria, es decir, se desplazaría más lento.
7. Para velocidades más altas del robot es necesario aumentar la magnitud de la desviación estándar, sin embargo, si la velocidad del móvil es muy alta, el algoritmo puede no converger debido a que el robot presenta movimientos más rápidos.
8. Aumentar el número de robots implica mayor costo computacional y por consecuencia un retardo mayor.

Como trabajo a futuro se mejorará el desempeño del algoritmo calculando la desviación estándar en función de los errores y de los retardados del sistema, de la forma:

$$\sigma_x = \gamma|x_e(t-\tau)| \quad \sigma_y = \gamma|y_e(t-\tau)| \quad \sigma_\theta = \gamma|\theta_e(t-\tau)| \tag{10}$$

Esto con el objetivo de reducir el campo de muestreo del filtro de partículas y tener una mejor aproximación de la estimación del estado.

REFERENCIAS

A. Morales, H. Nijmeijer, H.G.. (2013). A coordination control strategy for a group of unicycle robots. In *Congreso anual Asociación de México de Control Automático*.

Duan, Z., Cai, Z., and Yu, J. (2007). Robust position tracking for mobile robots with adaptive evolutionary particle filter. In *Natural Computation, 2007. ICNC 2007. Third International Conference on*, volume 4, 563–567. doi:10.1109/ICNC.2007.641.

Hsu, C.C., Wong, C.C., Teng, H.C., Li, N.J., and Ho, C.Y. (2010). Localization of mobile robots via an enhanced particle filter. In *Instrumentation and Measurement Technology Conference (I2MTC), 2010 IEEE*, 323–327. doi:10.1109/IMTC.2010.5488234.

Kawai, Y., Namerikawa, T., and Fujita, M. (2010). Bilateral teleoperation of wheeled mobile robot with time delay using virtual image robot. In *Control Applications (CCA), 2010 IEEE International Conference on*, 77–82. doi:10.1109/CCA.2010.5611232.

Kim, J., Kim, J., and Kim, D. (2011). Development of an efficient obstacle avoidance compensation algorithm considering a network delay for a network-based autonomous mobile robot. In *Information Science and Applications (ICISA), 2011 International Conference on*, 1–9. doi:10.1109/ICISA.2011.5772431.

Rusdinar, A., Kim, J., and Kim, S. (2010). Error pose correction of mobile robot for slam problem using laser range finder based on particle filter. In *Control Automation and Systems (ICCAS), 2010 International Conference on*, 52–55.

Sira-Ramírez, H., López-Uribe, C., and Velasco-Villa, M. (2010). Trajectory-tracking control of an input delayed omnidirectional mobile robot. In *Electrical Engineering Computing Science and Automatic Control (CCE), 2010 7th International Conference on*, 470–475. doi:10.1109/ICEEE.2010.5608660.

Wardhana, A., Clearesta, E., Widyotriatmo, A., and Suprijanto (2013). Mobile robot localization using modified particle filter. In *Instrumentation Control and Automation (ICA), 2013 3rd International Conference on*, 161–164. doi:10.1109/ICA.2013.6734064.

Yin, B., Wei, Z., Cong, Y., and Xu, T. (2010). A novel particle filter method for mobile robot localization. In *Measuring Technology and Mechatronics Automation (ICMTMA), 2010 International Conference on*, volume 1, 269–272. doi:10.1109/ICMTMA.2010.32.

Yu, J., Tang, Y., Cai, Z., and Duan, Z. (2008). Monte carlo localization for mobile robot with the improvement of particle filter. In *Intelligent Control and Automation, 2008. WCICA 2008. 7th World Congress on*, 3910–3914.