

## Terrestrial Paths Estimation by using Satellite Tracking with lost data

C.A. Rivera-Romero \* Ro. Olivera-Reyna \* O. Vite-Chávez \*  
J. Flores-Troncoso \*\* J.U. Muñoz-Minjares \*

\* *Unidad Académica de Ingeniería Eléctrica Plantel Jalpa, Universidad  
Autónoma de Zacatecas, 99601, Zac., México.*

\*\* *Unidad Académica de Ingeniería Eléctrica, Universidad Autónoma  
de Zacatecas, Av. Ramón López Velarde, 801, Col. Centro, 98000  
Zacatecas, México*

---

### Abstract:

Autonomous vehicles navigation requires previously plotted paths to avoid deviations or collisions with other objects. One of the main problems is the loss of location data when specific paths are designed due to disturbances in the acquisition devices. The most used technology for geo-position registration is the Global Positioning System (GPS) due to its great precision and low cost. However, the advantages of this technique for designing navigation paths are overshadowed if there is a significant loss of information.

In this work, it is proposed to use a modified Iterative Unbiased Finite Impulse Response (I-UFIR) algorithm to estimate the GPS data lost. The satellite measurements are obtained with the G28U7FTTL receptor, and the lost measurements are estimated in different sections of a specific path in an urban environment. The results are corroborated by using a real map and the mean square error based on a referenced path.

*Keywords:* Navigation, Loss Data, GPS, I-UFIR, Estimation.

---

### 1. INTRODUCCIÓN

La navegación autónoma de vehículos para realizar diversas tareas ha sido uno de los problemas más retadores hoy en día. Este desafío se complica debido a la enorme cantidad de variables a procesar si no se posee una ruta predeterminada (Cui et al., 2021). Actualmente se han desarrollado diferentes metodologías y se han utilizado diferentes tecnologías modernas para resolver este problema (Li and Yang, 2021; Phueakthong and Varagul, 2021). Sin embargo, una de las alternativas más confiables es establecer una ruta previa de navegación del vehículo (Darias et al., 2021; Do Quang et al., 2022).

En particular, la orientación de vehículos autónomos en zonas urbanas requieren de sistemas de navegación con el mínimo error posible para evitar daños físicos. No obstante, los avances tecnológicos continúan combatiendo con las perturbaciones externas en los dispositivos electrónicos, que causan inestabilidad en las mediciones o fallas en la recolección de información (Zou et al., 2022; Grage et al., 2019). En vista de esto, se requiere de algoritmos capaces de eliminar el ruido en las mediciones

de localización y de estimar los datos perdidos.

En cuestión de localización de vehículos, el Sistema de Posicionamiento Global (GPS, por sus siglas en inglés), permite resolver los problemas de rastreo de dispositivos alrededor del mundo con alta precisión (Bajaj et al., 2002). Las comunicaciones GPS son gratis y de acceso universal con un amplio rango de aplicaciones incluyendo exploración, recreación, búsqueda y rescate, y recuperar vehículos robados. Así, debido a sus grandes ventajas la creación de rutas basadas en mediciones mediante satélites es mejor que otras tecnologías inalámbricas. Por esta razón, algunos vehículos diseñados para operar bajo condiciones en exteriores emplean los servicios del GPS (Kumar et al., 2020; Zhou et al., 2018). Algunos ejemplos de dispositivos controlados por GPS son descritos en Rahiman and Zainal (2013) y Akhshirsh et al. (2021).

Sin embargo, la precisión de localización del GPS depende del costo del dispositivo receptor. Un receptor de bajo costo produce un nivel alto de ruido en la medición, tales perturbaciones dificultan la navegación y localización de un objeto con una precisión aceptable (Sheng et al., 2020; Roberts et al., 2018; Ray et al., 2019). Aunado a esto, las

fallas en los sistemas electrónicos en la etapa de muestreo de las coordenadas satelitales causa pérdidas de datos en una ruta específica (Krenn et al., 2011).

Diversos algoritmos han sido desarrollados para eliminar el ruido y compensar las mediciones pérdidas de sistemas reales. El filtro Kalman (KF) es uno de los más utilizados debido a su nivel de optimización, alta precisión y bajo costo computacional (Zhou et al., 2021). Sin embargo, este algoritmo a pesar de ser óptimo requiere información acerca del proceso y del ruido. Debido a que esta información no siempre se encuentra disponible, el algoritmo KF presenta una debilidad asociada con errores extra en algunas aplicaciones (Pomárico-Franquíz and Shmaliy, 2014). Contrariamente a los filtros KF, algunos estimadores operan con mediciones finitas y poseen una alta robustez (Zhao et al., 2014; Shmaliy, 2009). Recientemente, un filtro iterativo sin Sesgo de Respuesta Finita al Impulso (I-UFIR) fue desarrollado en Shmaliy et al. (2017) para demostrar un funcionamiento superior que el filtro Kalman en aplicaciones del mundo real.

Este algoritmo es apropiado para un amplio rango de aplicaciones prácticas, porque puede operar ignorando las estadísticas del ruido y las condiciones iniciales del sistema. Además, en el trabajo realizado en Uribe-Murcia et al. (2019), se propuso una modificación al algoritmo I-UFIR para compensar las mediciones pérdidas. Debido a las ventajas del filtro I-UFIR, se propone evaluar el desempeño del mismo utilizando mediciones propias mediante señales de referencia creadas por interpolación. Realizando un análisis de error medio cuadrático respecto a las señales de referencia, se puede determinar el error del filtro I-UFIR. Así, este algoritmo puede ser explotado para estimar las rutas requeridas por vehículos autónomos en zonas urbanas basada en mediciones ruidosas o con ausencia de información de un receptor GPS de bajo costo.

Este trabajo está organizado de la siguiente manera. Una breve descripción del filtro I-UFIR que compensa la ausencia de mediciones se presenta en la Sección 2. Después, se describe brevemente las características del equipo utilizado para adquirir las mediciones GPS descritas en la Sección 3, así como del lugar donde se realizaron las pruebas. Finalmente, se presenta la estimación de la ausencia de datos y el análisis de errores de la misma en las secciones de Resultados y Conclusiones (Sección 4 y Sección 5).

## 2. ESTIMADOR I-UFIR

Como primer paso, se describe el Modelado en Espacio de Estados para compensar la ausencia de mediciones de un proceso definido, como es el caso de mediciones satelitales y el Algoritmo I-UFIR detallado. Además, se

presenta el pseudo-algoritmo de manera minuciosa para su implementación.

### 2.1 Modelado en Espacio de Estados

Un algoritmo para estimar las pérdidas y retrasos de información fue diseñado en el espacio de estados en Uribe-Murcia et al. (2019). En este escenario, para retardos discretos  $h_k \geq 0$  el espacio de estados puede ser formalizado de la siguiente manera:

$$\mathbf{x}_k = \mathbf{A}\mathbf{x}_{k-1} + \mathbf{B}w_k, \quad (1)$$

$$\bar{\mathbf{y}}_k = \mathbf{C}\mathbf{A}\mathbf{x}_{k-1}, \quad (2)$$

$$\mathbf{y}_k = \kappa_k \mathbf{C}\mathbf{x}_{k-h_k} + (1 - \kappa_k)\bar{\mathbf{y}}_k + v_k, \quad (3)$$

donde  $\mathbf{x}_k \in \mathbb{R}^K$  es el vector de estado,  $\mathbf{y}_k \in \mathbb{R}^M$  es el vector de observación,  $\mathbf{A} \in \mathbb{R}^{K \times K}$ , es la matriz de estados,  $\mathbf{B} \in \mathbb{R}^{K \times P}$ , y  $\mathbf{C} \in \mathbb{R}^{M \times K}$ .

En la ecuación (3)  $\kappa_k$  es un escalar que representa el factor de datos perdidos. Si los datos arriban con éxito, con o sin retardo, se representan con  $\kappa_k = 1$ . Así, la ecuación (3) se reescribe como  $\mathbf{y}_k = \mathbf{C}\mathbf{x}_{k-h_k} + v_k$ . Ahora, si no existe retardo de las mediciones, en este caso  $h_k = 0$  y la ecuación (3) se reescribe como  $\mathbf{y}_k = \kappa_k \mathbf{C}\mathbf{x}_k + (1 - \kappa_k)\bar{\mathbf{y}}_k + v_k$ , el cual será el caso de estudio de este trabajo. Los vectores de ruido blanco gaussiano  $w_k \sim \mathcal{N}(0, Q_k) \in \mathbb{R}^P$  y  $v_k \sim \mathcal{N}(0, R_k) \in \mathbb{R}^M$  son considerados de media cero,  $E\{w_k\} = 0$  and  $E\{v_k\} = 0$ , con las covarianzas  $E\{w_k w_n^T\} = Q_k \delta_{k-n}$  y  $E\{v_k v_n^T\} = R_k \delta_{k-n}$  con la propiedad de  $E\{w_k v_n^T\} = 0$  para todo  $n$  y  $k$ .

Basándose en las ecuaciones (1) y (3), el algoritmo del filtro I-UFIR puede ser diseñado como a continuación se describe.

### 2.2 Algoritmo Iterativo UFIR

El filtro iterativo sin sesgo de respuesta finita al impulso (I-UFIR), es una conversión recursiva del filtro UFIR en formato de bloques, el cual es obtenido mediante dos etapas: predicción y actualización en Shmaliy et al. (2017). Este algoritmo no requiere conocimiento previo de las estadísticas del ruido ni los valores iniciales del proceso. El algoritmo I-UFIR requiere de la estimación del estado previo

$$\hat{\mathbf{x}}_l^- = \mathbf{A}\hat{\mathbf{x}}_{l-1} \quad (4)$$

para el estado pasado conocido  $\hat{\mathbf{x}}_{n-1}$  y así no contemplar el error de covarianza previo. Para la etapa de actualización, el algoritmo UFIR recalcula la ganancia generalizada de potencia del ruido (GNPG, por sus siglas en inglés)  $\mathbf{G}_l$  como:

$$\mathbf{G}_l = [\mathbf{C}^T \mathbf{C} + (\mathbf{A} \mathbf{G}_{l-1} \mathbf{A}^T)^{-1}]^{-1}, \quad (5)$$

tal ecuación se deriva de  $\mathbf{G}_l = (\mathbf{W}_{m,l}^T \mathbf{W}_{m,l})^{-1}$  donde  $\mathbf{W}_{m,l}$  es la ganancia del filtro UFIR (Shmaliy et al., 2017).

Los residuos de las mediciones pueden ser definidos como:

$$z_l = y_l - \mathbf{C} \hat{\mathbf{x}}_l^-, \quad (6)$$

la corrección de la ganancia de sesgo

$$\mathbf{K}_l = \mathbf{G}_l \mathbf{C}^T, \quad (7)$$

y el estado estimado

$$\hat{\mathbf{x}}_l = \hat{\mathbf{x}}_l^- + \mathbf{K}_l z_l. \quad (8)$$

Si se obtiene la estimación  $\hat{\mathbf{x}}_l$ , el filtrado puede calcularse estableciendo el valor de  $q = 0$  en la siguiente matriz del sistema:

$$\hat{\mathbf{x}}_{l-q} = \mathbf{A}^{-q} \hat{\mathbf{x}}_l. \quad (9)$$

Así, el estado filtrado puede obtenerse de la siguiente manera:

$$\hat{\mathbf{x}}_l = \mathbf{I} \hat{\mathbf{x}}_l. \quad (10)$$

El pseudo-código del algoritmo iterativo de filtrado UFIR desarrollado en Uribe-Murcia et al. (2019) se describe en el Algoritmo 1. El algoritmo presenta algunas modificaciones ya que solo se contempla el caso de pérdida de datos. Básicamente, esta modificación reduce el tiempo de procesamiento al eliminar la etapa del retraso de información.

---

**Algorithm 1** Pseudo-algoritmo de Filtro Iterativo para pérdida de datos.

---

**Data:**  $y_k, N, \kappa_k$

**Result:**  $\hat{\mathbf{x}}_k$

```

1: Begin :
2: for  $k = N - 1, N, \dots$  do
3:    $m = k - N + 1, s = k - N + K$ 
4:   if  $\text{isnan}(y_k)$  then
5:      $y_k = \mathbf{C} \mathbf{A} \mathbf{x}_{k-1}$ 
6:   end if
7:    $\mathbf{G}_s = (\mathbf{W}_{m,s}^T \mathbf{W}_{m,s})^{-1}$ 
8:    $\tilde{\mathbf{x}}_s = \mathbf{G}_s \mathbf{W}_{m,s}^T \mathbf{Y}_{m,s}$ 
9:   for  $l = s + 1$  to  $k$  do
10:     $\tilde{\mathbf{x}}_l^- = \mathbf{A} \tilde{\mathbf{x}}_{l-1}$ 
11:     $\mathbf{G}_l = [\mathbf{C}^T \mathbf{C} + (\mathbf{A} \mathbf{G}_{l-1} \mathbf{A}^T)^{-1}]^{-1}$ 
12:     $\mathbf{K}_l = \mathbf{G}_l \mathbf{C}^T$ 
13:     $\tilde{\mathbf{x}}_l = \tilde{\mathbf{x}}_l^- + \mathbf{K}_l (y_l - \mathbf{C} \tilde{\mathbf{x}}_l^-)$ 
14:   end for
15:    $\hat{\mathbf{x}}_k = \tilde{\mathbf{x}}_k$ 
16: end for
17: † Los datos  $y_0, y_1, \dots, y_{N-1}$  deben estar disponibles.

```

---

### 3. ADQUISICIÓN DE DATOS GPS

Las mediciones de GPS se obtuvieron mediante un prototipo ensamblado y configurado en la Universidad Autónoma de Zacatecas. Este dispositivo utiliza un receptor G28U7FTTL, una placa de desarrollo basada en el microcontrolador Atmega328P y un adaptador de memoria SD. El registrador de coordenadas satelitales es capaz de censar mediciones de latitud, longitud y altura en metros sobre el nivel del mar con una frecuencia de registro de una muestra por segundo. Este dispositivo es capaz de registrar las coordenadas de latitud y longitud de cualquier vehículo que pueda soportar el peso del dispositivo GPS, cuya masa es de 0.2 kilogramos.

Las mediciones fueron realizadas en el centro de la ciudad de Jalpa, Zacatecas alrededor de una manzana. Se eligió esta localización para lograr observar algunas perturbaciones en las mediciones ya que existen diversas señales de ruido ambiental. Las mediciones fueron corroboradas mediante el software Google Maps. En este trabajo se mostrará el procesamiento de las coordenadas por separado para analizar los errores. Esto debido a que latitud y longitud no presentan las mismas proporciones en las escalas en la ruta registrada.

### 4. RESULTADOS

#### 4.1 Estimación de ausencia de datos

Los datos registrados en la ubicación descrita en 3, fueron obtenidos con pocas pérdidas de datos pero con una cantidad significativa para el tamaño de la ruta especificada. Los datos fueron convertidos a coordenadas  $X$  y  $Y$  para su procesamiento y visualización utilizando las herramientas publicadas en Piretzidis and Sideris (2016). Cabe resaltar que los datos perdidos son almacenados como datos  $NaN$  "Not a Number" por sus siglas en inglés.

Se obtuvieron 223 muestras de la ruta especificada y los números de datos perdidos por intervalo fueron de 6, 10, 8 y 7. Estos intervalos pueden observarse en las Figuras 1 y 2 delimitados por círculos. En las mismas figuras se muestran las mediciones en puntos grises mientras el filtrado de la señal se representa con una línea sólida, respectivamente. Se puede apreciar en las Figuras 1 y 2 como la estimación I-UFIR logra adaptarse a la señal a pesar de no existir información en esos lapsos de la señal adquirida.

Durante las primeras pérdidas de la señal que representa las coordenadas  $Y$ , ver Figura 2, se puede apreciar una respuesta abrupta por parte del filtro I-UFIR, mientras que esto no sucede durante el procesamiento de las coordenadas  $X$ . Esto se debe a los cambios repentinos de la señal de rastreo y veremos posteriormente como afecta

este fenómeno a la señal estimada cuando se representa en un mapa real.

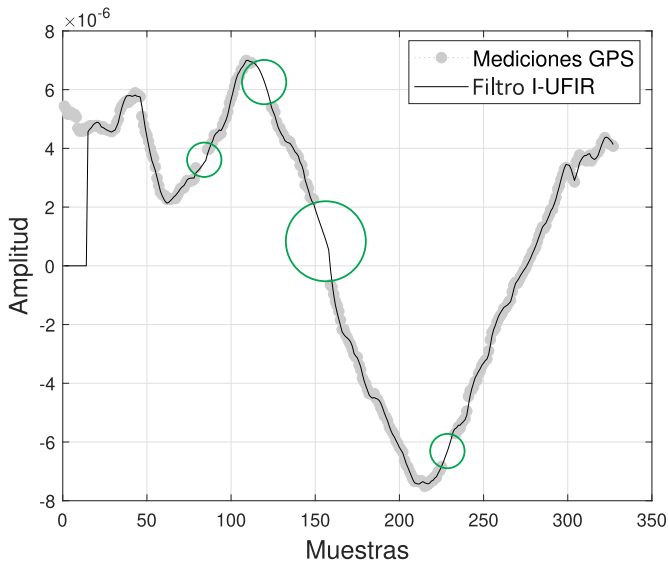


Fig. 1. Medición GPS. Coordenadas  $X$ , estimación e intervalos de pérdida de información.

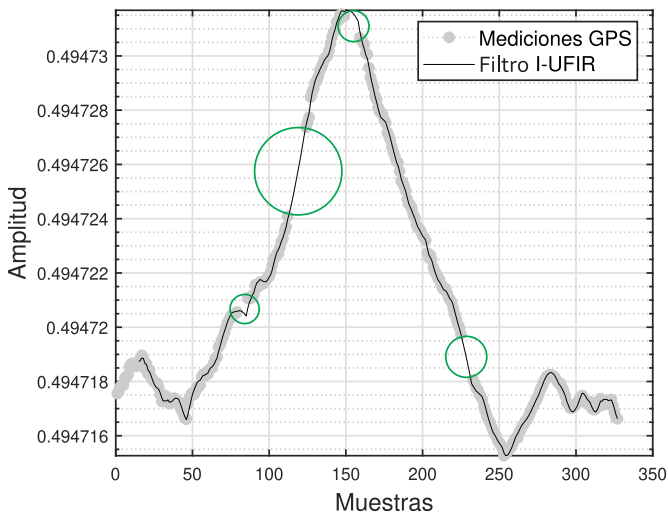


Fig. 2. Medición GPS. Coordenadas  $Y$ , estimación e intervalos de pérdida de información.

Una vez obtenido el filtrado de las mediciones, se procede a validarlos en un mapa real mediante la herramienta de Google Maps. En la Figura 3 se puede observar las mediciones obtenidas mediante puntos negros y el filtrado mediante una línea sólida. Aquí, pueden ser apreciados los límites de las calles y la manzana recorrida. En la Figura 3 podemos observar diferentes eventos:

- En el primer cambio señalado con la letra  $I_1$  presenta un cambio abrupto en el filtrado de la señal.

- Algunas mediciones están fuera de los límites de las calles.
- Durante el segundo intervalo señalado con la letra  $I_2$  la estimación de la ruta se obtuvo de manera correcta, sin embargo, la señal resultante invade la cuadra en cuestión.
- Los cambios  $I_3$  e  $I_4$  son estimados de manera correcta dentro de los límites de las calles.

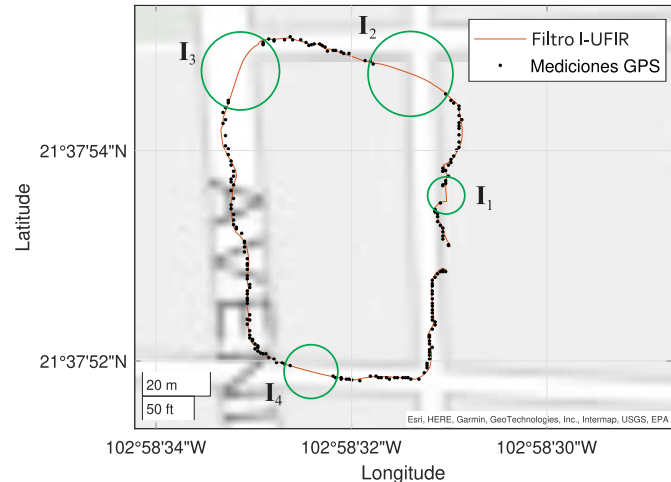


Fig. 3. Mediciones GPS y filtrado de mediciones con datos perdidos.

#### 4.2 Análisis de Errores

En la Sección 4.1 se mostraron las rutas estimadas a pesar de la pérdida de datos satelitales durante el recorrido del dispositivo GPSUAZ1.0 en un entorno urbano. Aquí, se corroboró la estimación de manera visual con la herramienta Google Maps. Sin embargo, un análisis numérico de errores se debe realizar para evaluar las estimaciones obtenidas. Dado que no se tiene una señal de referencia para tal evaluación, se realizó una interpolación de las muestras obtenidas usando dos técnicas: Pchip y Makima. La técnica Makima (Akima, 1970) está basada en una modificación del algoritmo Akima la cual se basa en la interpolación de Hermite por partes. Mientras que el algoritmo Pchip (Fritsch and Carlson, 1980) realiza una estimación simple basada en el polinomio de interpolación de Hermite cúbico por partes.

La interpolación de las coordenadas  $X$  e  $Y$  se muestran en las Figuras 4 y 5. En estas figuras se muestran las interpolaciones Pchip y Makima mediante una línea sólida y una línea punteada, respectivamente. Se puede observar en las Figuras 4 y 5 que durante las pérdidas de información no existen cambios abruptos de amplitud. Además, la interpolación se ve beneficiada al tener la información completa de las muestras para procesar la señal. Mientras

que el algoritmo I-UFIR realiza un filtrado de los datos generando así un mayor error. Es decir, el algoritmo de filtrado realiza un procesamiento en tiempo real una vez que se obtuvo el rango de mediciones  $N$ . Por lo tanto, las interpolaciones pueden ser utilizadas como referencia para calcular el error de la estimación del filtro I-UFIR.

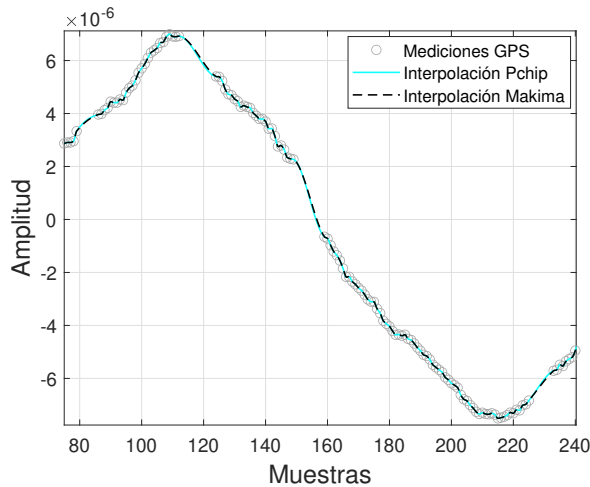


Fig. 4. Mediciones GPS en coordenadas cartesianas para el eje X, e interpolaciones usando Pchip y Makima.

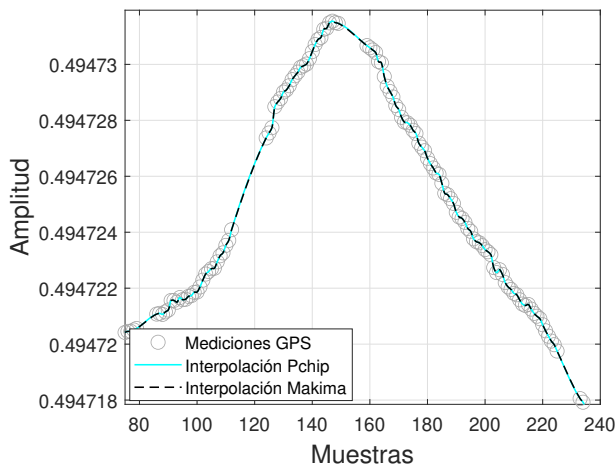


Fig. 5. Mediciones GPS en coordenadas cartesianas para el eje Y, e interpolaciones usando Pchip y Makima.

En la Tabla 1 se muestran los errores de estimación durante los trayectos donde se pierden los datos satelitales. El error del filtro I-UFIR con respecto a las interpolaciones propuestas es mínimo entre niveles de exponentes de  $e^{-13}$  y  $e^{-15}$ . Las diferencias mínimas son alcanzadas utilizando las mediciones de la técnica Makima, excepto para el eje Y en el intervalo  $I_3$  y para el intervalo  $I_4$ . El valor mínimo y máximo generado entre el filtro I-UFIR y las interpolaciones esta señalado en la tabla con los

símbolos  $\downarrow$  y  $\uparrow$ , respectivamente.

Table 1. Error medio cuadrático ( $mse$ ) de la estimación I-UFIR y las interpolaciones Pchip y Makima.

Lapso	Coordenada	Pchip	Makima
$I_1$	X	$1.1861e^{-13}$	$1.0169e^{-13}$
	Y	$1.2704e^{-13}$	$1.2106e^{-13}$
$I_2$	X	$8.2120e^{-14}$	$5.0159e^{-14}$
	Y	$1.0330e^{-13}$	$8.7442e^{-14}$
$I_3$	X	$3.6695e^{-13} \uparrow$	$3.0564e^{-13}$
	Y	$2.0493e^{-13}$	$2.1679e^{-13}$
$I_4$	X	$3.9808e^{-15} \downarrow$	$6.6977e^{-15}$
	Y	$2.3037e^{-14}$	$2.7038e^{-14}$

## 5. CONCLUSIONES

En este trabajo se comprobó la habilidad del receptor G28U7FTTL para obtener las coordenadas satelitales en entornos urbanos. Los datos recabados fueron corroborados al graficarlos mediante la herramienta libre Google Maps y observar que están dentro de la ruta recorrida con algunos errores debido al margen de error del GPS civil.

El algoritmo I-UFIR logró filtrar las mediciones y compensar los datos perdidos durante el trayecto planteado. A pesar de que algunos trayectos presentaban una pérdida de 10 datos aproximadamente, se logró estimar la ruta propuesta. Gracias a la interpolación de las técnicas Pchip y Makima se calculó el error medio cuadrático aproximado de la estimación durante los periodos de pérdida de señal GPS. El menor orden de los errores para el eje X y Y fue de  $e^{-15}$  para ambos casos. Así, se verificó que el filtro I-UFIR puede recuperar los datos perdidos con gran precisión. Para trabajo a futuro se pretende implementar en hardware el algoritmo I-UFIR, aunque se debe considerar que el tamaño del horizonte  $N$  puede incrementar significativamente el procesamiento de los datos.

## REFERENCES

- Akhshirsh, G.S., Al-Salihi, N.K., and Hamid, O.H. (2021). A cost-effective gps-aided autonomous guided vehicle for global path planning. *Bulletin of Electrical Engineering and Informatics*, 10(2), 650–657.
- Akima, H. (1970). A new method of interpolation and smooth curve fitting based on local procedures. *Journal of the ACM (JACM)*, 17(4), 589–602.
- Bajaj, R., Ranaweera, S.L., and Agrawal, D.P. (2002). Gps: location-tracking technology. *Computer*, 35(4), 92–94.
- Cui, H., Fan, X., and Li, Y. (2021). Research on state estimation of outdoor mobile robot based on multiple sensor combination. In *2021 IEEE 2nd International*

- Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)*, 907–910. IEEE.
- Darias, H., López-Estrada, F., Valencia-Palomo, G., Brindis-Velazquez, O., and Dominguez-Zenteno, J. (2021). Object transportation using cooperative differential mobile robots. *Memorias del Congreso Nacional de Control Automático*.
- Do Quang, H., Le Tran, T., Manh, T.N., Manh, C.N., Nhu, T.N., and Duy, N.B. (2022). Design a nonlinear mpc controller for autonomous mobile robot navigation system based on ros. *International Journal of Mechanical Engineering and Robotics Research*, 11(6).
- Fritsch, F.N. and Carlson, R.E. (1980). Monotone piecewise cubic interpolation. *SIAM Journal on Numerical Analysis*, 17(2), 238–246.
- Grage, T., Schoemann, M., Kieslich, P.J., and Scherbaum, S. (2019). Lost to translation: How design factors of the mouse-tracking procedure impact the inference from action to cognition. *Attention, Perception, & Psychophysics*, 81(7), 2538–2557.
- Krenn, P.J., Titze, S., Oja, P., Jones, A., and Ogilvie, D. (2011). Use of global positioning systems to study physical activity and the environment: a systematic review. *American journal of preventive medicine*, 41(5), 508–515.
- Kumar, M., Sarim, M., and Nemati, A. (2020). Autonomous navigation and target geo-location in gps denied environment. In *Multi-Rotor Platform-based UAV Systems*, 153–175. Elsevier.
- Li, M. and Yang, C. (2021). Navigation simulation of autonomous mobile robot based on teb path planner. In *Proceedings of the 2021 International Conference on Control and Intelligent Robotics*, 687–691.
- Phueakthong, P. and Varagul, J. (2021). A development of mobile robot based on ros2 for navigation application. In *2021 International Electronics Symposium (IES)*, 517–520. IEEE.
- Piretzidis, D. and Sideris, M.G. (2016). Map-lab: A matlab graphical user interface for generating maps for geodetic and oceanographic applications. In *Poster presented at the international symposium on gravity, geoid and height systems*, 19–23.
- Pomárico-Franquíz, J.J. and Shmaliy, Y.S. (2014). Accurate self-localization in rfid tag information grids using fir filtering. *IEEE Transactions on Industrial Informatics*, 10(2), 1317–1326.
- Rahiman, W. and Zainal, Z. (2013). An overview of development gps navigation for autonomous car. In *2013 IEEE 8th Conference on Industrial Electronics and Applications (ICIEA)*, 1112–1118. IEEE.
- Ray, J.D., Vijayan, M., and Kumar, A. (2019). Noise characteristics of gps time series and their influence on velocity uncertainties. *Journal of Earth System Science*, 128(6), 1–14.
- Roberts, G.W., Tang, X., and He, X. (2018). Accuracy analysis of gps/bds relative positioning using zero-baseline measurements. *The Journal of Global Positioning Systems*, 16(1), 1–8.
- Sheng, C., Fan, G., Yu, B., Gan, X., and Zhang, J. (2020). Optimized algorithm for bds/gps rtk suitable for urban canyons with a low-cost receiver. *Measurement Science and Technology*, 31(11), 115007.
- Shmaliy, Y. (2009). *GPS-based optimal FIR Filtering of Clock Models*. Nova Science Publishers.
- Shmaliy, Y.S., Zhao, S., and Ahn, C.K. (2017). Unbiased finite impulse response filtering: An iterative alternative to kalman filtering ignoring noise and initial conditions. *IEEE Control Systems Magazine*, 37(5), 70–89.
- Uribe-Murcia, K.J., Shmaliy, Y.S., Ahn, C.K., and Zhao, S. (2019). Unbiased fir filtering for time-stamped discretely delayed and missing data. *IEEE Transactions on Automatic Control*, 65(5), 2155–2162.
- Zhao, S., Shmaliy, Y.S., and Liu, F. (2014). Fast computation of discrete optimal fir estimates in white gaussian noise. *IEEE Signal Processing Letters*, 22(6), 718–722.
- Zhou, P., Zhang, S., Wen, L., Fu, J., Chai, T., and Wang, H. (2021). Kalman filter-based data-driven robust model-free adaptive predictive control of a complicated industrial process. *IEEE Transactions on Automation Science and Engineering*.
- Zhou, S., Miao, Z., Liu, Z., Zhao, H., Wang, H., Chen, H., and Liu, Y.H. (2018). Vision-based state estimation and trajectory tracking control of car-like mobile robots with wheel skidding and slipping. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 4270–4275. IEEE.
- Zou, Z., Huang, J., and Luo, P. (2022). Compensation tracker: Reprocessing lost object for multi-object tracking. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 307–317.