

Navigation System for MACÁBOT an Autonomous Surface Vehicles Using GPS Aided Strapdown Inertial Navigation System

B. Menna, *Member, IEEE*, S. Villar, *Member, IEEE*, and G. Acosta, *Senior Member, IEEE*

Abstract—In this work the design, implementation and real-time tests of a navigation system for the autonomous surface vehicle MACÁBOT is presented. This vehicle represents a versatile platform to perform several tasks in the marine environment, such as; ports maintenance, marine productive ecosystems studies and bathymetries. The navigation system is responsible for accurately determining the position, velocity and attitude of the vehicle. It represents a fundamental component to autonomously carry out any of the aforementioned tasks. In this work, the navigation system is developed based on a GPS aided strap-down inertial navigation system using an extended Kalman filter sensor fusion algorithm. In order to provide an adaptive approach to the sensor fusion algorithm tuning a fuzzy inference system is used. The navigation system was implemented as a package for the Robot Operating System, benefiting from the advantages of heterogeneity, integration and hardware abstraction. Real time tests of the MACÁBOT on a local creek were carried out, showing satisfactory performance of the navigation system in both position and velocity estimates. In addition to these tests, simulations of GPS outages were carried out with the registered data to evaluate the performance of the navigation system in such cases.

Index Terms—Autonomous Surface Vehicles, Aided Inertial Navigation System, Extended Kalman Filter, Fuzzy Inference System, Robot Operating System.

I. INTRODUCCIÓN

En el campo de la robótica móvil autónoma marina, los Vehículos Autónomos de Superficie (ASVs, Autonomous Surface Vehicles) representan plataformas versátiles que permiten abordar diversas aplicaciones en entornos de aguas someras marinas, hidrográficas, lacustres, y portuarias. Estas aplicaciones consisten en: el monitoreo del estado de

infraestructura portuaria y canales de acceso, estudio de ecosistemas productivos marinos, prospecciones hidrográficas, entre otras [1]–[6]. En esta línea y para solventar algunas de las aplicaciones mencionadas, en los laboratorios del Núcleo INTELYMEC (Investigación Tecnológica en Electricidad y Mecatrónica) – CIFICEN (Centro de Investigaciones en Física e Ingeniería del Centro), de la Universidad Nacional del Centro de la Provincia de Buenos Aires (UNCPBA), Argentina se ha desarrollado y construido un ASV, denominado MACÁBOT.

El ASV-MACÁBOT, como cualquier robot móvil autónomo, debe desempeñar en forma autónoma su misión teniendo la capacidad de planificar trayectorias seguras y eficientes a través de un entorno desconocido. Para ello, se requiere de una arquitectura de Navegación, Guiado y Control (NGC, Navigation, Guidance and Control) [7]–[8], como así también de una re-planificación dinámica de los objetivos de la misión. En general, la arquitectura NGC está compuesta de: Sistema de Navegación (NS, Navigation System), Sistema de Guiado (GS, Guidance System), Sistema de Control (CS, Control System) y Sistema Planificador de Misión Dinámico (DMPS, Dynamic Mission Planning System). El NS determina la posición, velocidad y orientación del móvil en el entorno donde desarrolla su misión. El GS genera las consignas necesarias para establecer las trayectorias que debe seguir el móvil, partir de conjuntos de *waypoints*, proporcionados por el DMPS. El CS comprende los lazos de realimentación para el control del movimiento del móvil en función de las consignas generadas por el GS. Además, se requiere de un DMPS para ajustar los objetivos de la misión en función de las capacidades del ASV o de limitaciones impuestas por el entorno [9]–[10]. Esto completa un sistema de control jerárquico de tiempo real.

La información determinada por el NS es una componente fundamental para el funcionamiento de este sistema. El NS se implementa típicamente utilizando una Unidad de Medición Inercial (IMU, Inertial Measurement Unit), montada sobre el móvil, que proporciona mediciones de aceleraciones y velocidades angulares. Si se utilizan IMUs de grado de navegación, es posible implementar el NS resolviendo un conjunto de ecuaciones de navegación [11], solución conocida como sistema de navegación inercial adherido (SDINS, Strapdown Inertial Navigation System). En cambio, para IMUs de grado medio más económicas, pero con especificaciones de error inferiores, la solución de dichas

Bruno V. Menna, INTELYMEC-CIFICEN-CONICET, Dto. de Ing. Electromecánica, Facultad de Ingeniería, Universidad Nacional del Centro de la Prov. de Bs. As. Olavarría - Argentina, bruno.menna@fio.unicen.edu.ar.

Sebastián A. Villar, INTELYMEC-CIFICEN-CONICET, Dto. de Ing. Electromecánica, Facultad de Ingeniería, Universidad Nacional del Centro de la Prov. de Bs. As. Olavarría - Argentina, svillar@fio.unicen.edu.ar.

Gerardo G. Acosta, INTELYMEC-CIFICEN-CONICET, Dto. de Ing. Electromecánica, Facultad de Ingeniería, Universidad Nacional del Centro de la Prov. de Bs. As. Olavarría - Argentina, ggacosta@fio.unicen.edu.ar.

ecuaciones de navegación acumula con el tiempo un error creciente y no acotado en las estimaciones de velocidad y posición [12]. Esto se debe a que para resolver las ecuaciones de navegación se realiza una doble integración de las aceleraciones medidas por la IMU, por lo que cualquier error en estas se acumula con el tiempo.

Una solución que permite el empleo de IMUs de grado medio consiste en la utilización de instrumentos de asistencia. Este enfoque se conoce como SDINS asistido por dispositivos (D-A-SDINS, **D**evices-**A**ided-**S**trapdown **I**NS) [13]. Algunos de los instrumentos de asistencia utilizados en robótica móvil autónoma incluyen; odómetros, sistema de posicionamiento global y diferencial (GPS, **G**lobal **P**ositioning **S**ystem y DGPS, **D**ifferential **G**PS), compás electrónico, registradores de velocidad Doppler (DVL, **D**oppler **V**elocity **L**oggers), transponders acústicos, altímetros, entre otros [14]–[17]. Un D-A-SDINS se desempeña correctamente mientras disponga de mediciones de los instrumentos de asistencia, de lo contrario, el rendimiento empeora, pudiendo funcionar aceptablemente en dicha condición por un tiempo limitado.

Un D-A-SDINS requiere emplear algoritmos de fusión de sensores (SFA, **S**ensor **F**usion **A**lgorithm) para corregir el error del SDINS a partir de las mediciones de asistencia. Entre las técnicas de fusión de sensores usuales se tienen el Filtro de Kalman Extendido (EKF, **E**xtended **K**alman **F**ilter) [18], el Filtro de Partículas (PF, **P**article **F**ilter) [19] y Redes Neuronales Artificiales (ANN, **A**rtificial **N**eural **N**etwork) [20].

El EKF representa una alternativa válida para el desarrollo inicial del NS del ASV-MACÁBOT. Por ser un algoritmo ampliamente probado. Aunque menos preciso y robusto respecto a implementaciones basadas en PF, su costo computacional es menor [21], mientras que respecto a soluciones basadas en ANN, dependerá de la arquitectura de la red, del tipo de ANN y de si requiere reentrenamiento en línea [22]. Sin embargo, el correcto funcionamiento del EKF requiere la sintonía de parámetros de ruido tanto del proceso a estimar como de las mediciones. En este sentido, son varios los enfoques que se han aplicado, como los basados en la innovación (IAE, **I**nnovation-based **A**daptive **E**stimation) [23], empleando ANN [24], Algoritmos Evolutivos (EA, **E**volutionary **A**lgorithms) [25] y Sistemas de Inferencia Difusos (FIS, **F**uzzy **I**nferece **S**ystems) [26]. De estos enfoques, los basados en FIS proveen resultados apropiados para la aplicación de este trabajo con un costo computacional bajo.

Mediante este trabajo se pretende presentar el desarrollo, implementación y prueba de una solución de navegación basada en un SDINS asistido por GPS (GPS-A-SDINS), empleando un EKF como SFA y un FIS para la sintonía adaptiva de sus parámetros, apropiada para su empleo en móviles autónomos, en este caso el ASV MACÁBOT. Si bien la técnica de fusión de sensores basada en el EKF está ampliamente probada y documentada, se brindan detalles de implementación de valor para otros trabajos. Esto se refuerza mediante el empleo del Sistema Operativo de Robótica (ROS, **R**obot **O**perating **S**ystem) para la implementación,

evidenciando las ventajas que este brinda para la integración de sensores y sistemas en plataformas móviles y la posibilidad de recolección de datos para análisis posteriores. A su vez, al ser ROS una iniciativa de distribución libre, propicia el intercambio de desarrollos como el presentado en este trabajo para su empleo por parte de otros grupos.

El trabajo se encuentra organizado de la siguiente manera: en la Sección II se describen las características generales del ASV-MACÁBOT y sus principales componentes en términos de hardware y software; en la Sección III, se fundamenta el NS desarrollado; la Sección IV detalla la implementación del NS en ROS; en la Sección V se presentan los resultados experimentales; y en la Sección VI se discuten las conclusiones obtenidas.

A fin de facilitar la lectura de este trabajo, en la Tabla I se listan los acrónimos más utilizados.

TABLA I
LISTADO DE ACRÓNIMOS UTILIZADOS EN ESTE TRABAJO

| Acrónimo | Significado |
|--------------------|--|
| <i>ASV</i> | Autonomous Surface Vehicle |
| <i>AUV</i> | Autonomous Underwater Vehicle |
| <i>CS</i> | Control System |
| <i>D-A-SDINS</i> | Device Aided Strap Down Inertial Navigation System |
| <i>DCM</i> | Direction Cosine Matrix |
| <i>DMPS</i> | Dynamic Mission Planning System |
| <i>DOM</i> | Degree Of Mismatch |
| <i>EKF</i> | Extended Kalman Filter |
| <i>FIS</i> | Fuzzy Inference System |
| <i>GPS-A-SDINS</i> | GPS Aided Strap Down Inertial Navigation System |
| <i>GS</i> | Guidance System |
| <i>LIN</i> | Local Interconnect Network |
| <i>NED</i> | North East Down |
| <i>NGC</i> | Navigation Guidance and Control |
| <i>NS</i> | Navigation System |
| <i>ROS</i> | Robot Operating System |
| <i>SDINS</i> | Strap Down Inertial Navigation System |
| <i>SFA</i> | Sensor Fusion Algorithm |

II. PLATAFORMA ASV-MACÁBOT

El ASV-MACÁBOT, cuyo nombre hace referencia al Macá, un ave acuática sudamericana, está constituido por un chasis de acero inoxidable que integra dos flotadores de triple capa de PVC, como se aprecia en la Fig. 1. Sobre el chasis se montan una caja estanca que contiene la electrónica, y dos cajas estancas más pequeñas, para alojar las baterías que alimentan la electrónica y el sistema de propulsión. El sistema de propulsión lo constituyen propulsores eléctricos de corriente continua del tipo usado en embarcaciones ligeras. Operan a 12 V y proveen al vehículo un empuje aproximado de 98 N, desarrollando una velocidad máxima sostenida de 1 m/s. Son manejados mediante controladores tipo puente H de 20 A de corriente nominal. Este sistema permite obtener una potencia acorde a las aplicaciones en entornos de aguas someras y el empleo de dos propulsores permite establecer el rumbo mediante propulsión diferencial, sin la necesidad de un timón.

Al chasis se acopla un soporte con una quilla de aluminio cuya profundidad puede variarse, mostrada en la Fig. 2, donde

se montan diferentes sensores como SONAR de barrido lateral, ecosonda mono-haz de barrido mecánico, SONAR de barrido frontal, MODEM acústico y otros sensores adicionales necesarios para el desarrollo de la misión. La quilla está aislada del chasis para evitar corrosión galvánica.

Estas características lo convierten en una plataforma multipropósito adaptable a las aplicaciones ya mencionadas.



Fig. 1. ASV-MACÁBOT, destacándose los flotadores de PVC y las cajas estancas para electrónica y baterías montadas sobre el chasis de acero inoxidable.



Fig. 2. Sensores acústicos montados en la quilla (arriba) consistentes en: SONAR de barrido lateral StarFish, ecosonda Super SeaKing y SONAR de barrido frontal Micron, equipos de la firma Trittech® (abajo, de izquierda a derecha).

A. Componentes de Hardware

La electrónica del ASV se diseñó en dos sistemas de procesamiento principales: a) de alto nivel y b) de bajo nivel.

El sistema de alto nivel lo constituye una CPU Advantech® ARK-1380, con procesador Intel Celeron 1 GHz, de un consumo aproximado de 35 W, y SSD de 64 Gb. A la CPU se conectan directamente por puertos serie universales los dispositivos GPS e IMU. También se conectan a sus puertos serie RS-232 los sensores montados en la quilla (ver Fig. 2).

El sistema de bajo nivel consiste de una configuración maestro-esclavos sobre un bus LIN (Local Interconnect Network). Los nodos conectados sobre el bus LIN emplean microcontroladores MSP430F247 de Texas Instruments® como procesador central. El nodo maestro se encarga de las comunicaciones entre los nodos esclavos y el sistema de alto nivel. Los nodos esclavos desarrollados y operativos a la fecha son; interfaces con los controladores de motores, gestión de energía y de emergencia. Los subsistemas de alto y bajo nivel se comunican vía puerto serie universal.

Como dispositivos de navegación posee una IMU y un receptor de GPS. La IMU empleada es marca Xsens®, modelo MTI30-2A5G4 AHRS (Attitude and Heading Reference System) [27]. Proporciona mediciones de aceleraciones, velocidades angulares, de campo magnético y una estimación de la orientación en cuaterniones o ángulos de Euler a una tasa de 50 Hz en este caso. El GPS empleado es marca Garmin®, modelo 18xLVC, un receptor GPS compacto y con antena integrada [28]. Proporciona datos de navegación GPS en tramas NMEA0183 a una tasa de 1Hz.

B. Arquitectura de Software

La programación del ASV-MACÁBOT refleja los lineamientos establecidos para la robótica móvil autónoma, que fueron presentados (ver Sección I). Su implementación fue realizada en ROS, un sistema operativo de distribución libre que se ejecuta sobre Linux, instalado en la CPU del ASV. El sistema ROS es ampliamente utilizado en la comunidad de robótica y consiste de un conjunto de procesos independientes, denominados *nodos*, ejecutados en paralelo y con la capacidad de intercambiar *mensajes* entre ellos. El intercambio de mensajes se realiza a través de *tópicos*. De manera que un nodo puede publicar mensajes en un tópico o recibirlos suscribiéndose a un tópico [29].

El diseño de la arquitectura de software desarrollada se beneficia de las múltiples características y ventajas proporcionadas por ROS. Entre las que se destacan: la capacidad de encapsular nodos desarrollados en diferentes lenguajes de programación, la incorporación inmediata y homogénea de nuevos nodos de procesamiento o para manejo de nuevos dispositivos, tanto sensores como actuadores, aún para dispositivos con drivers propietarios y muy diferentes entre sí y el cómputo distribuido. Todo esto facilita el crecimiento incremental y el prototipado del sistema.

III. NS PROPUESTO PARA EL ASV-MACÁBOT

El NS desarrollado consiste de un SDINS, empleando una IMU de grado medio, asistido por GPS, es decir, un GPS-A-SDINS. El SDINS resuelve un conjunto de ecuaciones de navegación [30] para calcular la velocidad y posición a partir de las mediciones de aceleración y orientación obtenidas de la IMU. Sin embargo, como se mencionó antes, debido a los errores en las mediciones de aceleración de la IMU, los resultados obtenidos con el SDINS acumulan error con el tiempo. Estos errores son corregidos a partir de las mediciones de asistencia proporcionadas por el GPS, empleando un SFA basado en un EKF del tipo *loosely integration* [31]. Para el correcto funcionamiento del EKF se requiere la sintonía de un conjunto de parámetros de covarianza de ruido, para esto se aplicó un FIS.

Las estimaciones de posición, velocidad y orientación se obtienen para un marco de referencia geográfico o n (n -frame) cuyos ejes se orientan según las direcciones norte, este y hacia abajo normal a la superficie, abreviado NED (North East Down), este marco de referencia queda definido por un plano tangente a la superficie de la Tierra en una posición de latitud L y longitud l , medidas respecto a un marco de

referencia centrado en la Tierra o e . Mientras que las aceleraciones del vehículo están referidas a un marco de referencia móvil y fijo al vehículo, también denominado del cuerpo rígido o b (b -frame) de ejes XYZ. La transformación de las aceleraciones del marco b al n para su integración se realizan mediante una rotación según la orientación del vehículo. Los marcos de referencia mencionados se representan en la Fig. 3.

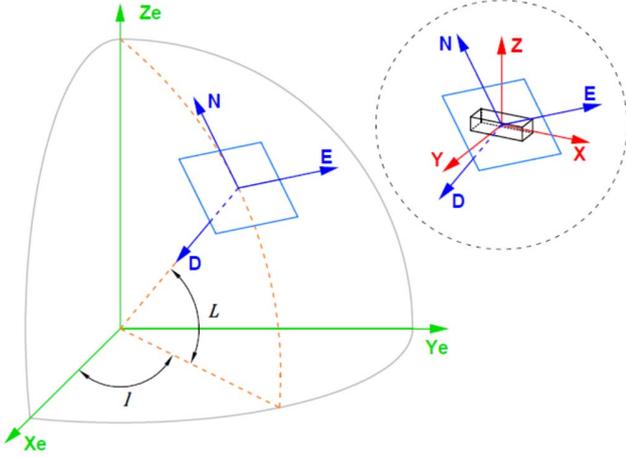


Fig. 3. Marcos de referencia de la Tierra (verde), geográfico (azul) y del cuerpo rígido (rojo).

Se describen a continuación los fundamentos involucrados en el planteamiento de un SDINS y posteriormente los vinculados al SFA y su aplicación al GPS-A-SDINS desarrollado.

El cambio de velocidad del móvil respecto al marco de referencia n se obtiene como se muestra a continuación:

$$\dot{V}^n = \begin{bmatrix} \dot{V}_N \\ \dot{V}_E \\ \dot{V}_D \end{bmatrix} = f^n - \Delta f^n - f_c^n + g^n \quad (1)$$

donde \dot{V}^n es el cambio de velocidad, \dot{V}_N , \dot{V}_E y \dot{V}_D representan los cambios de velocidad en las direcciones NED, f^n son las aceleraciones referidas al marco n , Δf^n representa los errores en las aceleraciones, f_c^n es la aceleración de Coriolis y g^n agrupa las aceleraciones de la gravedad y centrípeta.

Las aceleraciones medidas por la IMU se encuentran en el marco de referencia b y deben ser transformadas, mediante una rotación, al marco de referencia n para obtener medidas globales:

$$f^n = \begin{bmatrix} f_N \\ f_E \\ f_D \end{bmatrix} = C_b^n f^b = C_b^n \begin{bmatrix} f_X \\ f_Y \\ f_Z \end{bmatrix} \quad (2)$$

donde f^b representa las aceleraciones referidas al marco b y C_b^n es la Matriz de Cosenos Directores (DCM), cuyo computo se realiza según:

$$C_b^n = \begin{bmatrix} (a^2 + b^2 - c^2 - d^2) & 2(bc - ad) & 2(bd + ac) \\ 2(bc + ad) & (a^2 - b^2 + c^2 - d^2) & 2(cd - ab) \\ 2(bd - ac) & 2(cd + ab) & (a^2 - b^2 - c^2 + d^2) \end{bmatrix} \quad (3)$$

donde a , b , c y d son los parámetros de orientación del móvil en cuaterniones. Se eligió esta representación para evitar la singularidad conocida como "gimbal lock" dada en la representación de Euler para ángulos de cabeceo de $\pm\pi/2$. La DCM es una matriz ortonormal, lo que es verificado y corregido cada vez que se la computa [32].

La aceleración de Coriolis es calculada según:

$$f_c^n = \begin{bmatrix} 2\omega_e V_E \sin(L) - \frac{V_N V_D - V_E^2 \tan(L)}{R_0 + h} \\ 2\omega_e (V_N \sin(L) + V_D \cos(L)) - \frac{V_E (V_D + V_D \tan(L))}{R_0 + h} \\ 2\omega_e V_E \cos(L) + \frac{V_E^2 + V_N^2}{R_0 + h} \end{bmatrix} \quad (4)$$

donde ω_e la velocidad de rotación de la Tierra, R_0 el radio de la Tierra, L la latitud, h la altitud y V_N , V_E y V_D son las componentes de velocidad en las direcciones NED.

Las aceleraciones de la gravedad y centrípeta se calculan según:

$$g^n = \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} - \frac{\omega_e^2 (R_0 + h)}{2} \begin{bmatrix} \sin(2L) \\ 0 \\ 1 + \cos(2L) \end{bmatrix} \quad (5)$$

donde g es el valor local de la aceleración de la gravedad.

A partir de (1) se determinan las velocidades del cuerpo rígido respecto al marco n por integración.

El cambio de posición en el marco de referencia n se obtiene según:

$$\dot{p}^n = \begin{bmatrix} \dot{L} \\ \dot{l} \\ \dot{h} \end{bmatrix} = \begin{bmatrix} V_N / (R_0 + h) \\ V_E \sec(L) / (R_0 + h) \\ -V_D \end{bmatrix} \quad (6)$$

donde \dot{p}^n representa el cambio de posición y \dot{L} , \dot{l} y \dot{h} son los cambios en latitud, longitud y altitud, respectivamente.

A partir de (6) la posición del cuerpo en el marco n se obtiene por integración.

Las ecuaciones (1) y (6) asumen la forma de la Tierra esférica y no consideran variación de su campo gravitacional.

El conjunto de variables de salida del NS se agrupó en un vector de estados, expresado de la siguiente manera:

$$\hat{\mathbf{x}}_k = \left[\hat{L}_k \ \hat{l}_k \ \hat{h}_k \ \hat{V}_{Nk} \ \hat{V}_{Ek} \ \hat{V}_{Dk} \ \Delta \hat{f}_{Nk} \ \Delta \hat{f}_{Ek} \ \Delta \hat{f}_{Dk} \right]^T \quad (7)$$

donde $\hat{\mathbf{x}}_k$ es el vector de estados al instante k , \hat{L}_k , \hat{l}_k y \hat{h}_k son la latitud, longitud y altitud estimadas, \hat{V}_{Nk} , \hat{V}_{Ek} y \hat{V}_{Dk} son las velocidades estimadas en el marco n y $\Delta \hat{f}_{Nk}$, $\Delta \hat{f}_{Ek}$ y $\Delta \hat{f}_{Dk}$ representan las correcciones de las aceleraciones en el marco n .

La IMU empleada incorpora un SFA propietario para la estimación de la orientación. Para esto, integra las mediciones de velocidad angular de los giróscopos, cuyo error es corregido por el SFA utilizando una estimación de la orientación obtenida a partir de las mediciones de aceleraciones y campo magnético. Las especificaciones de error típico de orientación declaradas por el fabricante de la IMU van de 0.2° a 1° [27]. Mediante pruebas estáticas para distintas orientaciones conocidas se validaron dichas especificaciones. Concluyendo que los datos de orientación proporcionados por la IMU no requerían correcciones. Por esto no fueron incluidos en el vector de estados (7).

Notar que los estados de altitud \hat{h}_k y velocidad \hat{V}_{Dk} son incluidos también dentro de (7). Estos estados serán útiles en un futuro desarrollo para emplear el NS en un Vehículo Autónomo Submarino (AUV, Autonomous Underwater Vehicle). En ese caso se requerirá de dichos estados y nuevos instrumentos de asistencia para su corrección.

Se empleó un EKF para corregir la estimación de los estados [33]. El algoritmo recursivo EKF consta de dos etapas, predicción y corrección. En la etapa de predicción, las variables de estado se propagan según el modelo del proceso SDINS (1) y (6). La etapa de corrección se realiza cuando se dispone de mediciones válidas del GPS. Las variables de estado obtenidas se retroalimentan a la etapa de predicción en la siguiente iteración.

La etapa de predicción del EKF se computa resolviendo las siguientes ecuaciones:

$$\hat{\mathbf{x}}_k^- = \Gamma(\hat{\mathbf{x}}_{k-1}, \mathbf{f}_{k-1}^n) \quad (8)$$

$$\mathbf{P}_k^- = \mathbf{A}_k \mathbf{P}_{k-1} \mathbf{A}_k^T + \mathbf{Q}_k \quad (9)$$

donde $\hat{\mathbf{x}}_k^-$ es el vector de estados estimado, $\hat{\mathbf{x}}_{k-1}$ es el último vector de estados corregido, \mathbf{f}_{k-1}^n es el último vector de entrada, formado por las mediciones de aceleración referidas al marco n (2), la función no lineal $\Gamma(\cdot)$ representa la propagación de los estados. \mathbf{P}_k^- es la matriz de covarianza de

error predicha, \mathbf{A}_k es el Jacobiano de $\Gamma(\cdot)$ respecto de los estados, \mathbf{P}_{k-1} es la última matriz de covarianza de error corregida y \mathbf{Q}_k es la matriz de covarianza de ruido del proceso.

La etapa de corrección se computa resolviendo las siguientes ecuaciones:

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R})^{-1} \quad (10)$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{z}_k - \Upsilon(\hat{\mathbf{x}}_k^-)) \quad (11)$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^- \quad (12)$$

donde \mathbf{K}_k es la matriz de ganancia de Kalman, \mathbf{H}_k es el Jacobiano de $\Upsilon(\cdot)$ con respecto a los estados y \mathbf{R} es la matriz de covarianza de ruido de las mediciones de asistencia. El vector \mathbf{z}_k está formado por las mediciones de asistencia provistas por el GPS, $\Upsilon(\cdot)$ es una función que convierte los estados predichos a mediciones equivalentes y la matriz \mathbf{I} es una matriz identidad.

En la ecuación (11), la diferencia entre las mediciones de asistencia y las mediciones estimadas a partir de los estados predichos se denomina innovación:

$$\text{INN}_k = \mathbf{z}_k - \Upsilon(\hat{\mathbf{x}}_k^-) \quad (13)$$

Las matrices de las etapas de predicción y corrección se determinan utilizando:

$$\hat{\mathbf{x}}_k^- = \begin{bmatrix} \hat{p}_k^{n-} \\ \hat{V}_k^{n-} \\ \Delta \hat{f}_k^{n-} \end{bmatrix} = \begin{bmatrix} (\hat{p}_{k-1}^n + \dot{p}_k^n dT) \\ (\hat{V}_{k-1}^n + \dot{V}_k^n dT) \\ \Delta \hat{f}_{k-1}^n \end{bmatrix} \quad (14)$$

$$\mathbf{z}_k = [L_{GPSk} \ l_{GPSk} \ h_{Ak} \ V_{NGPSk} \ V_{EGPSk} \ V_{DAk}]^T \quad (15)$$

$$\Upsilon(\hat{\mathbf{x}}_k^-) = [\hat{L}_k^- \ \hat{l}_k^- \ \hat{h}_k^- \ \hat{V}_N^- \ \hat{V}_E^- \ \hat{V}_D^-]^T \quad (16)$$

$$\mathbf{P}_0 = (9 \times 9) \text{Identidad} \quad (17)$$

$$\mathbf{Q}_k = (9 \times 9) \text{Diagonal} \quad (18)$$

$$\mathbf{I} = (9 \times 9) \text{Identidad} \quad (19)$$

$$A_k = \begin{bmatrix} 0 & 0 & \frac{\partial L}{\partial h} & \frac{\partial L}{\partial V_N} & 0 & 0 & 0 & 0 & 0 \\ \frac{\partial l}{\partial L} & 0 & \frac{\partial l}{\partial h} & 0 & \frac{\partial l}{\partial V_E} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ \frac{\partial V_N}{\partial L} & 0 & \frac{\partial V_N}{\partial h} & \frac{\partial V_N}{\partial V_N} & \frac{\partial V_N}{\partial V_E} & \frac{\partial V_N}{\partial V_D} & -1 & 0 & 0 \\ \frac{\partial V_E}{\partial L} & 0 & \frac{\partial V_E}{\partial h} & \frac{\partial V_E}{\partial V_N} & \frac{\partial V_E}{\partial V_E} & \frac{\partial V_E}{\partial V_D} & 0 & -1 & 0 \\ \frac{\partial V_D}{\partial L} & 0 & \frac{\partial V_D}{\partial h} & \frac{\partial V_D}{\partial V_N} & \frac{\partial V_D}{\partial V_E} & \frac{\partial V_D}{\partial V_D} & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (20)$$

$$H_k = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad (21)$$

$$R = \begin{bmatrix} 5E-8 & 0 & 0 & 0 & 0 & 0 \\ 0 & 5E-8 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1.0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.25 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.25 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1.0 \end{bmatrix} \quad (22)$$

Para el correcto funcionamiento del EKF se requiere determinar las matrices de covarianzas de ruido del proceso y de las mediciones de asistencia Q_k y R .

La matriz R se determinó a partir del análisis estadístico de conjuntos de mediciones asistencia, obteniendo las varianzas de las medidas de posición y velocidad del GPS.

En cambio, la matriz Q_k , que representa la incertidumbre del modelo del proceso, no puede determinarse directamente. Por esto, en este trabajo se aplicó un proceso de adaptación de la matriz Q_k basado en la discrepancia entre las covarianzas de innovación teórica y observada [34], dadas respectivamente por:

$$C_{Tk} = H_k (A_k P_k^- A_k^T + Q_k) H_k^T + R \quad (23)$$

$$C_{Ok} = \frac{1}{N} \sum_i^k INN_k INN_k^T \quad (24)$$

La covarianza de innovación observada (24) se calcula para una ventana móvil de tamaño N , determinado experimentalmente, adoptando un valor de 4.

Se establece un factor de discrepancia (DOM, Degree Of Mismatch), dado por la diferencia entre las covarianzas de innovación teórica y observada, como se muestra en la siguiente ecuación:

$$DOM_k = C_{Tk} - C_{Ok} \quad (25)$$

En (23) se observa que Q_k es el único valor no determinado que puede influir en el valor de C_{Tk} . Así, en función del valor de DOM_k , se puede ajustar la incertidumbre del modelo del proceso, Q_k , para que la covarianza de innovación teórica coincida con la observada. De esta manera; si DOM_k es positivo, la covarianza de innovación observada es menor que la teórica, entonces el valor de Q_k debería reducirse. Si DOM_k es negativo, la covarianza de innovación observada es mayor que la teórica, entonces el valor de Q_k debería aumentarse. Mientras que si DOM_k es aproximadamente cero, Q_k no debería modificarse.

En base a estas tres reglas se implementó un FIS, que toma como entrada los valores de DOM_k y da como salida un factor α para ajustar la matriz Q_k . Respondiendo dicho ajuste a la siguiente ecuación:

$$Q_k = \alpha Q_k \quad (26)$$

Como entrada al FIS se empleó el promedio de los valores de DOM correspondientes a latitud y longitud.

En la Fig. 4 se representan los conjuntos difusos y funciones de pertenencia para entrada y salida. Se definieron cinco conjuntos de entrada para el valor de DOM_k ; NA (Negativo Alto), NB (Negativo Bajo), CE (Cero), PB (Positivo Bajo) y PA (Positivo Alto) y cinco conjuntos de salida para α ; DA (Decremento Alto), DB (Decremento Bajo), IC (Incremento Cero), IB (Incremento Bajo) e IA (Incremento Alto).

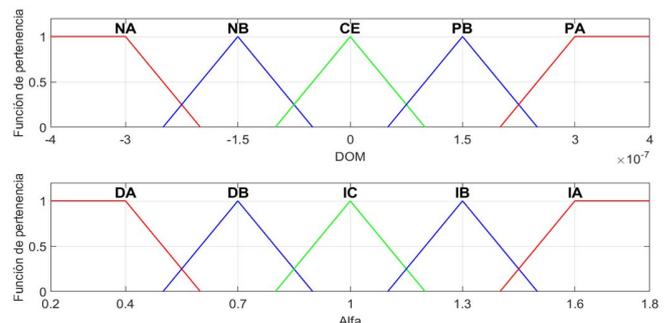


Fig. 4. Conjuntos difusos y funciones de pertenencia para la entrada (arriba) y salida (abajo).

La base de reglas la componen cinco reglas del tipo Mamdani:

1. *If*NA *Then* IA
2. *If*NB *Then* IB
3. *If*CE *Then* IC
4. *If*PB *Then* DB

5. *If*PA *Then* DA

Como método de defusificación se utilizó el de centroide.

En las Fig. 5 y 6 se muestran los diagramas de bloques para el algoritmo de SDINS y GPS-A-SDINS, respectivamente. En ambas figuras se referencian las ecuaciones involucradas y descritas anteriormente.

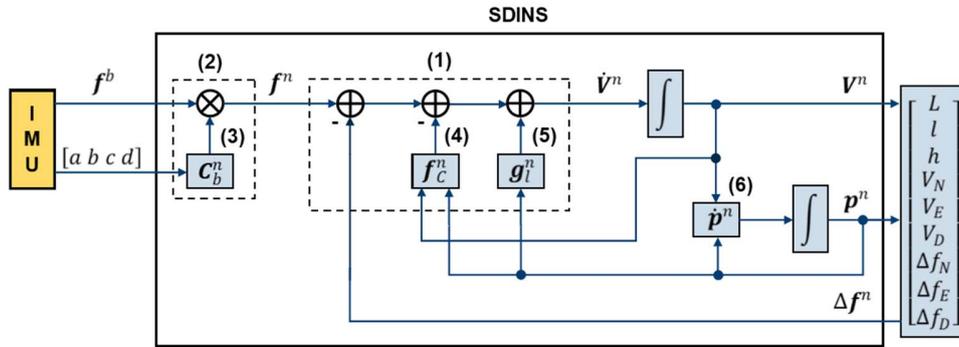


Fig. 5. Diagrama de bloques y ecuaciones del SDINS.

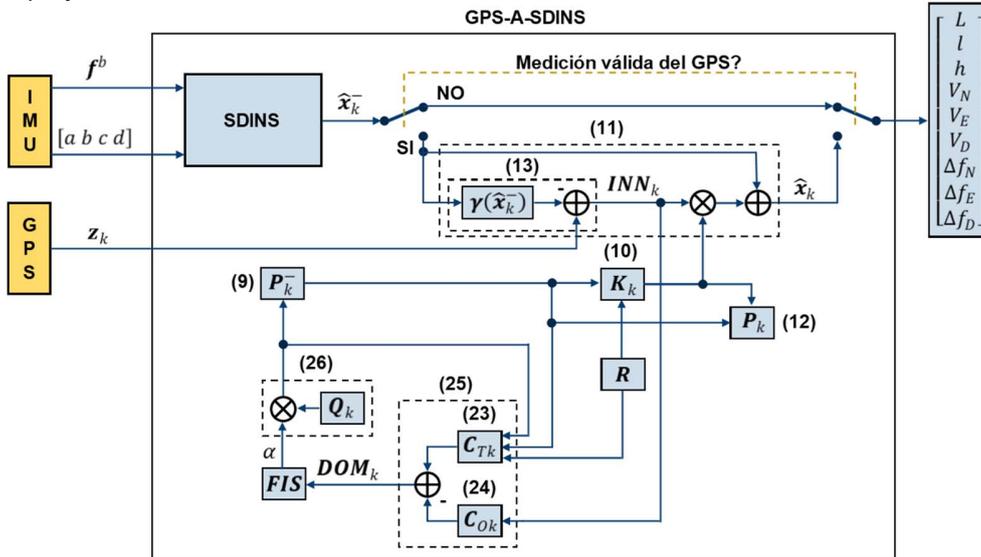


Fig. 6. Diagrama de bloques y ecuaciones del GPS-A-SDINS.

IV. IMPLEMENTACIÓN DEL NS EN EL ASV-MACÁBOT

El NS fue implementado como un paquete ROS, programado en lenguaje C++. Para su empleo en el ASV se instaló el paquete ROS del NS en la CPU Advantech® ARK-1380 portada por el ASV-MACÁBOT. Se utilizó como dispositivo de sensado para el SDINS una IMU Xsens® MTI30-2A5G4 [27] y como instrumento de asistencia un GPS Garmin® 18xLVC [28].

La Fig. 7 muestra un diagrama del diseño propuesto en ROS. En el diagrama, los rectángulos que contienen elipses corresponden a las estructuras *paquetes* y las elipses a los *nodos*, que son las instancias ejecutables de los paquetes. Los *tópicos*, el medio de comunicación entre nodos, se representan con rectángulos más pequeños. Las flechas entre nodos y tópicos hacen referencia a si el nodo es publicador (flecha

entrante al tópico) o suscriptor (flecha saliendo del tópico) a un tópico.

El nodo */imu_node* contenido en el paquete *imu_driver* corresponde al driver de la IMU Xsens® MTI30-2A5G4. Este nodo es proporcionado por Xsens® y permite la configuración del dispositivo y la publicación en el tópico */imu_output* de los mensajes de salida de la IMU a una tasa de 50 Hz.

El nodo */gps_node* contenido en el paquete *gps_driver* corresponde al driver del GPS Garmin® 18xLVC. Este nodo genérico es compatible con una amplia gama de dispositivos GPS con tramas en formato NMEA0183 estándar. Publica en el tópico */gps_output* datos de posición y velocidad a una tasa de 1 Hz.

En el nodo */ns_node* contenido en el paquete *ns_macabot* se implementó el NS. Se suscribe a los tópicos */imu_output* y */gps_output* para obtener las mediciones de la IMU y el GPS.

Publica en el t3pico `/ns_output` los datos de navegaci3n a una tasa de 50 Hz.

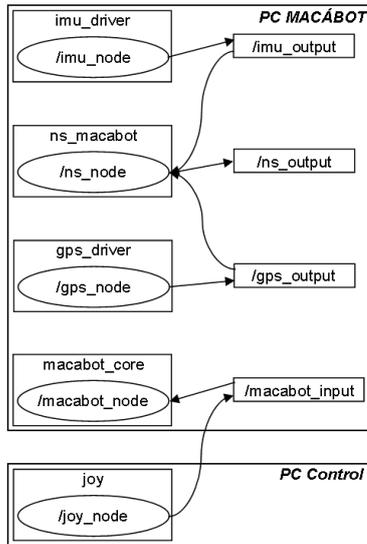


Fig. 7. Diagrama de los componentes de ROS incluyendo paquetes (rectángulos mas grandes), nodos (elipses), t3picos (rectángulos más pequeños) y flechas que indican publicaci3n o suscripci3n.

La implementaci3n del c3digo se realiz3 mediante dos clases, como se muestra en la Fig. 8.

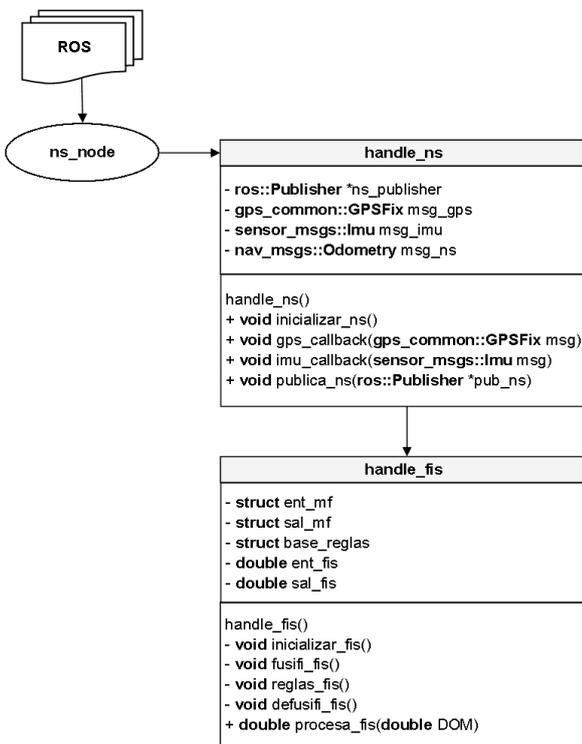


Fig. 8. Diagrama de clases, incluyendo los métodos implementados.

En la clase `handle_ns` se implement3 el SDINS y el EKF, posee los métodos; `inicializar_ns()` en el que se inicializan las variables empleadas, `gps_callback()` en el que se reciben los mensajes publicados por `/gps_node` y luego de verificar su

validez mediante el fix del GPS se carga el vector de mediciones de asistencia (15), `imu_callback()` en el que se reciben los mensajes publicados por `/imu_node`, con los datos de aceleraciones y orientaci3n, se computan los resultados del SDINS, que corresponden a la etapa de predicci3n del EKF. Luego, si se ha recibido un mensaje v3lido del GPS, se ejecuta la etapa de correcci3n del EKF y el FIS y `publica_ns()` en el que, a partir de los resultados obtenidos en `imu_callback()` se completa el mensaje de salida del nodo `/ns_node` y se lo publica en el t3pico `/ns_output`.

En la clase `handle_fis` se implement3 el FIS, posee los métodos: `inicializar_fis()` en el que se inicializan las variables empleadas, funciones de pertenencia de entrada y salida y base de reglas, `procesa_fis()` el cual es invocado desde el método `imu_callback()` en la etapa de correcci3n del EKF y con los valores de DOM (25) como argumento de entrada, este método realiza la inferencia difusa para obtener el factor de ajuste α de la matriz de covarianza de ruido del proceso. Para esto, se tienen los métodos privados; `fusifi_fis()` en el que se calcula el grado de pertenencia de la entrada, `reglas_fis()` evalúa la base de reglas y `defusifi_fis()` para obtener el valor de salida empleando las funciones de pertenencia de salida y desfusificador por centroide, el valor de salida obtenido es el factor α .

V. RESULTADOS EXPERIMENTALES

Para realizar las pruebas se desplegó el ASV-MACÁBOT en el arroyo Tapalque, ciudad de Olavarría. Se estableci3 una red de área local, empleando un `router` Wi-Fi, a la que se conectaron la PC del ASV y una PC para control del ASV mediante un `joystick` y recolecci3n de datos por medio de archivos `rosbags`.

Se presentan los resultados obtenidos para dos trayectorias. En ambos casos el ASV parti3 del reposo y con posici3n adquirida por el GPS, estableciendo las condiciones iniciales del NS. Las trayectorias elegidas presentan dos condiciones de navegaci3n en las que, para el caso de la trayectoria 1 predomina el desplazamiento del ASV sin cambios frecuentes de rumbo, mientras que en la trayectoria 2 los cambios de rumbo son más frecuentes y de 180°, similar a lo que se da cuando se realizan misiones de relevamiento hidrográfico. Es de interés analizar el desempeñio del NS en ambos casos. En las Fig. 9 y 10 se muestran las posiciones obtenidas con las mediciones del GPS y las estimadas por el NS implementado. En las Fig 11 y 12 las velocidades estimadas por el NS. Mientras que en las Fig. 13 y 14 el rumbo obtenido con las mediciones de la IMU.

Es v3lido aclarar que en las Fig. 11 y 12 se observan picos de velocidad superiores a 1 m/s, estos se deben a que, al no disponer de lazos de control de velocidad en el ASV, en los momentos en que se aplica máxima potencia con el `joystick` se manifiesta la respuesta con sobre-pico de la velocidad del veh3culo que supera la velocidad máxima final de 1m/s.

Estas pruebas se realizaron para determinar el desempeñio del NS en forma aislada, sin la interacci3n con los otros sistemas DMPS, GS y CS, que aportarían sus propios errores.

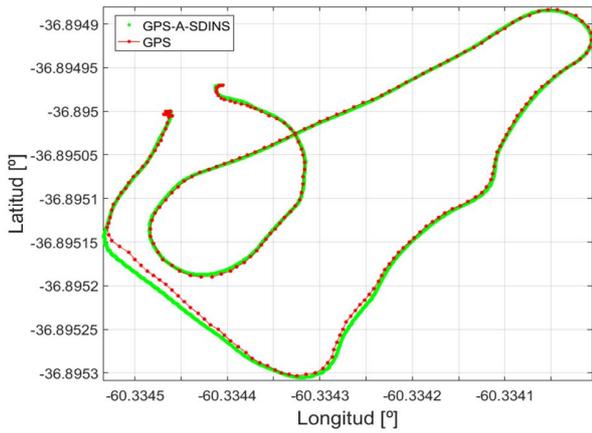


Fig. 9. Posiciones obtenidas por el GPS y el NS para la trayectoria 1.

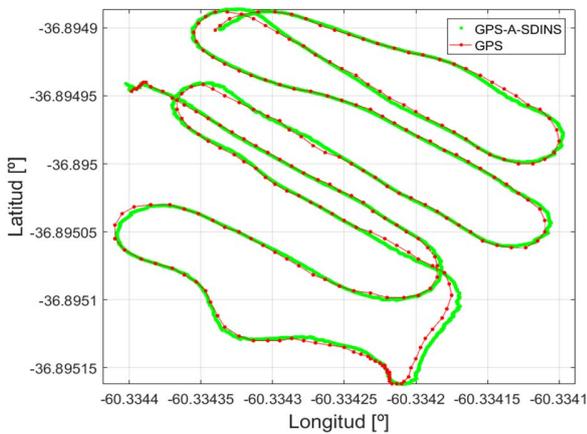


Fig. 10. Posiciones obtenidas por el GPS y el NS para la trayectoria 2.

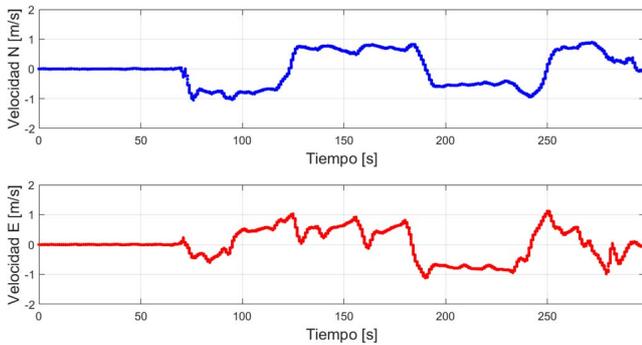


Fig. 11. Componentes de velocidad en direcciones Norte (arriba) y Este (abajo) obtenidas por el NS para la trayectoria 1.

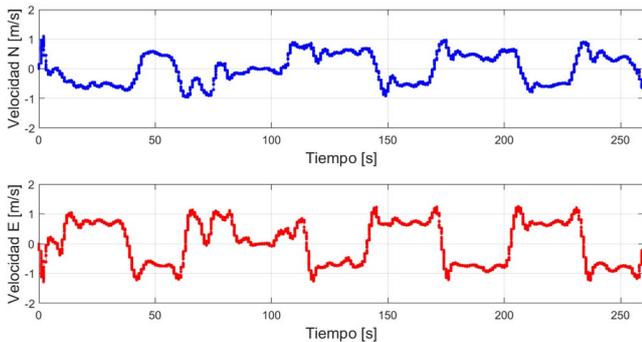


Fig. 12. Componentes de velocidad en direcciones Norte (arriba) y Este (abajo) obtenidas por el NS para la trayectoria 2.

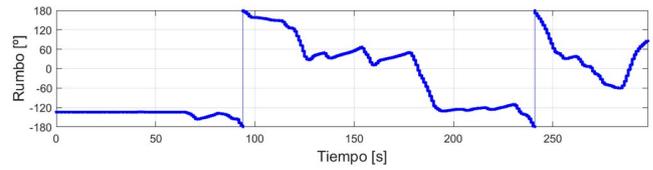


Fig. 13. Rumbo medido para la trayectoria 1.

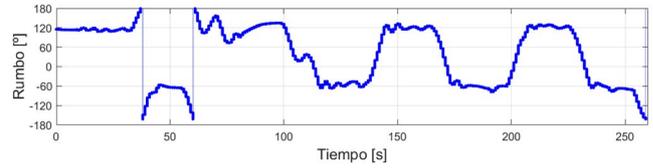


Fig. 14. Rumbo medido para la trayectoria 2.

Para llevar adelante este análisis del NS desarrollado hubiera sido deseable realizar una comparación con mediciones de un dispositivo de navegación de referencia, como por ejemplo un DGPS o un sistema de navegación comercial. Sin embargo, como actualmente no se dispone de dicho dispositivo de referencia, se realizó una comparación para las trayectorias completas entre las mediciones del GPS y los resultados del NS. El objetivo de esta comparación es demostrar que mediante el sistema desarrollado es posible obtener una solución de navegación con una tasa de actualización superior a la del GPS y a su vez con un error comparable. De este modo, el NS puede proporcionar valores de posición y velocidad en el intervalo de medidas disponibles de GPS, con un error comparable al de este último, asegurando una suerte de interpolación de suma utilidad para el correcto funcionamiento de los restantes sistemas.

Para determinar el error de posición entre las mediciones del GPS y las estimaciones del NS se emplearon las siguientes ecuaciones:

$$err_{pos} = 2R_0 \tan^{-1}(\sqrt{a}, \sqrt{1-a}) \quad (27)$$

$$a = \sin^2\left(\frac{L_{GPS} - L_{NS}}{2}\right) + \cos(L_{GPS}) \cos(L_{NS}) \sin^2\left(\frac{l_{GPS} - l_{NS}}{2}\right) \quad (28)$$

donde; err_{pos} es el error de posición en metros, R_0 es el radio de la Tierra en metros (6378137 m), L_{GPS} y l_{GPS} son las mediciones de latitud y longitud del GPS y L_{NS} y l_{NS} son las estimaciones de latitud y longitud del NS.

Los resultados de esta comparación se presentan en la Tabla II, en la que se muestran los valores de error medio cuadrático (RMS) de posición en dirección Norte, Este y horizontal (2D) y de velocidad en las direcciones Norte y Este.

| TABLA II ERRORES RMS DE POSICIÓN Y VELOCIDAD | | |
|---|---------------|---------------|
| Error | Trayectoria 1 | Trayectoria 2 |
| Posición Norte | 0.4031 m | 0.5927 m |
| Posición Este | 0.5257 m | 0.7618 m |
| Posición 2D | 0.6625 m | 0.9652 m |
| Velocidad Norte | 0.0499 m/s | 0.0789 m/s |
| Velocidad Este | 0.0634 m/s | 0.0885 m/s |

De los resultados obtenidos se comprueba que para ambas trayectorias el error en posición del NS respecto al GPS es inferior a un metro, mientras que el error de velocidad es, en ambos casos, inferior al 10% de la velocidad máxima de 1 m/s que desarrolla el ASV MACÁBOT.

Se observa que ambas trayectorias representan situaciones en las que se tienen disponibles mediciones del GPS a su frecuencia de refresco normal de 1 Hz. Para una evaluación del sistema ante una interrupción de las mediciones del GPS, se simularon para los datos registrados de las trayectorias 1 y 2, interrupciones en la recepción de datos GPS de distinta duración y en diferentes tramos de las trayectorias del ASV, que incluyen periodos en los que el ASV se encuentra detenido y en movimiento, incluyendo maniobras de giro. Esto resulta de interés para abordar casos más complejos de navegación en inmersión o ambientes sin posibilidad de uso de GPS, como es el caso de los AUV.

Los resultados obtenidos para el error de estimación de posición y velocidad para interrupciones del GPS de hasta 20 s en partes de las trayectorias en las que el ASV estaba detenido y en movimiento, que luego fueron promediadas se representan en las Fig. 15 y 16.

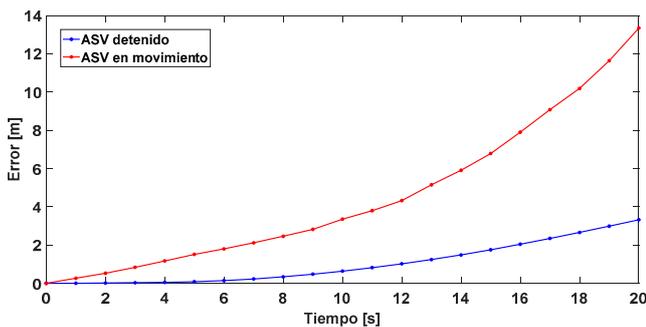


Fig. 15. Evolución del error en la estimación de posición del NS ante interrupciones en la recepción de datos GPS.

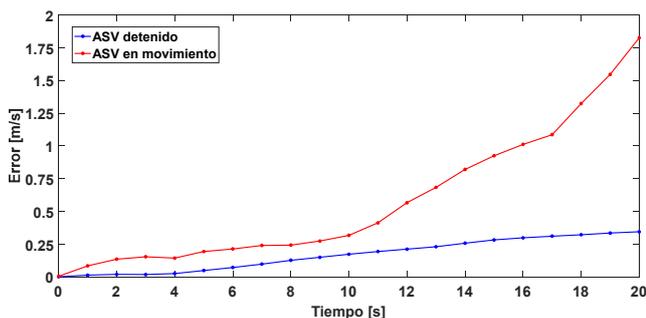


Fig. 16. Evolución del error en la estimación de velocidad del NS ante interrupciones en la recepción de datos GPS.

Se observa que mientras el ASV está detenido el error de estimación de posición es inferior al metro para interrupciones de hasta 12 s. Mientras que para el ASV en movimiento el crecimiento del error en la estimación de posición resulta más elevado. Si se considera que la exactitud del GPS empleado según especificaciones [28], en el mejor de los casos es inferior a los 3 m, para el caso del ASV en movimiento un error similar se tiene para interrupciones de duración inferior a

los 10 s, esta duración coincide además con un marcado incremento en el error de estimación de velocidad.

Considerando ambas situaciones puede establecerse como valor límite una interrupción en la recepción de GPS de hasta 10 s. Este valor debe ser considerado por el sistema planificador de misión DMPS del ASV para la toma de decisiones en caso de que se presente esta situación.

VI. CONCLUSIONES Y TRABAJO FUTURO

Debido a las inferiores especificaciones de error de las IMU de grado medio, como la empleada en el ASV-MACÁBOT, respecto a las de grado táctico o de navegación, la solución obtenida por un SDINS acumula error rápidamente. La precisión del NS representa una componente fundamental para que un móvil autónomo como el ASV-MACÁBOT lleve a cabo de forma autónoma una misión en particular. Por esto, en este trabajo se diseñó, implementó y probó en tiempo real y en forma aislada, un NS basado en un GPS-A-SDINS utilizando un EKF como algoritmo de fusión de sensores y sintonizado mediante FIS. Los resultados experimentales de navegación muestran que los datos obtenidos son adecuados para ser utilizados como realimentación en los sistemas de guiado, control y planificación dinámica del ASV, tanto en lo que respecta a la tasa de actualización como en la resolución de los mismos. Lo que constituye un objetivo importante en vistas a lograr la operación autónoma del ASV-MACÁBOT. A su vez, mediante la simulación de interrupciones en la recepción de datos GPS se pudo establecer un límite de tiempo máximo admisible para dicha interrupción.

La implementación del NS utilizando ROS demostró ser muy apropiada, especialmente en términos de heterogeneidad, abstracción de hardware, integración de sistemas y recolección de datos para análisis posteriores.

Se prevé continuar el desarrollo del NS. Estos trabajos incluyen el desarrollo en curso de una rutina de calibración de la IMU para compensar los posibles errores sistemáticos como *offsets*, errores de factor de escala y desalineación de los ejes de los sensores de la IMU, esto permitirá evaluar en qué proporción los errores de posición y velocidad del NS son debidos a los instrumentos y al algoritmo en sí. Adicionalmente se probarán otras técnicas para la sintonía del EKF. Así como su empleo como referencia para la evaluación de otras técnicas de fusión de sensores, en particular empleando filtros de partículas.

Además, el NS se ensayará para su empleo en un AUV [35], para lo cual se trabajará en la implementación de las medidas de asistencia necesarias, basadas en medición de distancias, ya que en el caso de un móvil marino sumergible no se contará con recepción de GPS cuando este se encuentre en inmersión.

REFERENCIAS

- [1] F. Arrichiello, H. K. Heidarsson, S. Chiaverini, and G. S. Sukhatme, "Cooperative caging and transport using autonomous aquatic surface vehicles," *Intelligent Service Robotics.*, vol. 5, no. 1, pp. 73–87, 2012.
- [2] R. Silva, S. Cunha, A. Matos and N. Cruz, "Shallow Water Surveying Using Experimental Interferometric Synthetic Aperture Sonar," *Marine Technology Society Journal*, vol. 43, no. 1, pp. 50-63, 2009.

- [3] M. Caccia, M. Bibuli, R. Bono, Ga. Bruzzone, Gi. Bruzzone and E. Spirandelli, "Unmanned Surface Vehicle for Coastal Protected Waters Applications: the Charlie Project," *Marine Technology Society Journal*, vol. 41, no. 2, pp. 62-71, 2007.
- [4] G. S. Sukhatme, A. Dhariwal, B. Zhang, C. Oberg, B. Stauffer, and D. A. Caron, "Design and development of a wireless robotic networked aquatic microbial observing system," *Environmental Engineering Science*, vol. 24, no. 2, pp. 205-215, 2007.
- [5] X. Lurton, *An Introduction to Underwater Acoustics. Principles and Applications*, Springer Praxis Books in Geophysical Sciences. Berlin Heidelberg: Springer, 2002.
- [6] L. Zhixiang, Z. Youmin, Y. Xiang, and Y. Chi, "Unmanned surface vehicles: An overview of developments and challenges," *Annual Reviews in Control*, pp. 71-93, 2016.
- [7] T. Fossen, *Marine Control Systems*, Marine Cybernetics, Norway, 2002.
- [8] T. Fossen, *Guidance and control of Ocean Vehicles*, Wiley, Chichester, UK, 1994.
- [9] G. G. Acosta, "State of the art on trajectory generation for AUV with artificial intelligence techniques," *AUVI Project Internal Report*, no. 3027-1, pp. 1-15, 2005.
- [10] G. G. Acosta, H. J. Curti, O. Calvo, and S. R. Rossi, "A knowledge-based approach for AUV path planner development," *WSEAS Transactions on Systems*, vol. 5, no. 6, pp. 1417-1424, 2006.
- [11] A. Ortyl and Z. Gosiewski, "Strapdown Inertial Navigation System Part 1 - Navigation Equations," *Journal of Theoretical and Applied Mechanics.*, vol. 36, no. 1, pp. 81-95, 1998.
- [12] P. Aggarwal, Z. Syed, A. Noureldin, and N. El-Sheimy, *MEMS-Based Integrated Navigation (GNSS Technology and Applications)*, 1st ed. Artech House, 2010.
- [13] J. F. Vasconcelos, C. Silvestre, and P. Oliveira, "INS/GPS aided by frequency contents of vector observations with application to autonomous surface crafts," *IEEE Journal of Oceanic Engineering*, vol. 36, no. 2, pp. 347-363, 2011.
- [14] D. D. S. Santana, C. M. Furukawa, and N. Maruyama, "Sensor Fusion with Low-Grade Inertial Sensors and Odometer to Estimate Geodetic Coordinates in Environments without GPS Signal," *IEEE LATIN AMERICA TRANSACTIONS*, vol. 11, no. 4, pp. 1015-1021, 2013.
- [15] S. Curi, I. Mas, and R. S. Peña, "Autonomous Flight of a Commercial Quadrotor," *IEEE LATIN AMERICA TRANSACTIONS*, vol. 12, no. 5, pp. 853-858, 2014.
- [16] L. Paull, S. Saecedi, M. Seto, and H. Li, "AUV Navigation and Localization - A Review," *IEEE Journal of Oceanic Engineering*, vol. 39, no. 1, pp. 1-25, 2014.
- [17] F. Kendoul, "Survey of Advances in Guidance, Navigation, and Control of Unmanned Rotorcraft Systems," *Journal of Field Robotics*, vol. 29, no. 2, pp. 315-378, 2012.
- [18] A. Kelly, "A 3D state space formulation of a navigation Kalman filter for autonomous vehicles," *Carnegie Mellon University Robotics Institute Technical Report*, no. 94-19, pp. 1-95, 1994.
- [19] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forszell, J. Jansson, R. Karlsson, and P.-J. Nordlund, "Particle Filters for Positioning, Navigation, and Tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 425-437, 2012.
- [20] A. M. Hasan, K. Samsudin, and R. Ramli, "GPS / INS Integration Based on Dynamic ANFIS Network," *International Journal of Control and Automation*, vol. 5, no. 3, pp. 1-22, 2012.
- [21] M. F. Bugallo, S. Xu, and P. M. Djurić, "Performance comparison of EKF and particle filtering methods for maneuvering targets," *Journal of Digital Signal Processing*, vol. 17, no. 4, pp. 774-786, 2007.
- [22] A. Shareef, Y. Zhu, M. Musavi, and B. Shen, "Comparison of MLP Neural Network and Kalman Filter for Localization in Wireless Sensor Networks," in *Proceedings of 19th IASTED International Conference on Parallel and Distributed Computing Systems*, pp. 323-330, 2007.
- [23] H. A. Mohamed and P. K. Schwarz, "Adaptive Kalman Filtering for INS/GPS," *Journal of Geodesy*, vol. 73, no. 4, pp. 193-203, 1999.
- [24] D. J. Jwo and H. C. Huang, "Neural network aided adaptive extended Kalman filtering approach for DGPS positioning," *Journal of Navigation*, vol. 57, no. 3, pp. 449-463, 2004.
- [25] D. Ji. Jwo and S. C. Chang, "Particle swarm optimization for GPS navigation Kalman filter adaptation," *Aircraft Engineering and Aerospace Technology.*, vol. 81, no. 4, pp. 343-352, 2009.
- [26] M. Liu and F. Xiong, "A Fuzzy Adaptive GPS/INS Integrated Navigation Algorithm," *Procedia Engineering.*, vol. 15, pp. 660-664, 2011.
- [27] Xsens Technologies, "MTi 10 and MTi 100 series Technical Specifications," 2014.
- [28] Garmin International, "GPS 18x Technical Specifications," 2011.
- [29] A. Koubaa, *Robot Operating System (ROS) The Complete Reference*, 1st. Ed., vol. 1. Springer International Publishing, 2016.
- [30] D. Titterton and J. Weston, *Strapdown Inertial Navigation Technology*, Institution of Engineering and Technology, 2004.
- [31] M. Grewal, L. Weill, and A. Andrews, *Global Positioning Systems, Inertial Navigation and Integration*, John Wiley and Sons, 2007.
- [32] I. Y. Bar-itzhack and K. A. Fegley, "Orthogonalization Techniques of a Direction Cosine Matrix," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 5, no. 5, pp. 798-804, 1969.
- [33] G. Welch and G. Bishop., "An Introduction to the Kalman Filter," University of North Carolina, Department of Computer Science, *Technical Report 95-041*, 1995.
- [34] D. J. Jwo and F. C. Chung, "Adaptive Kalman Filter for Navigation Sensor Fusion," *Sensor Fusion and its Applications*, T. Ciza, Sciyo, pp. 65-91, 2010.
- [35] G. G. Acosta, H. Curti, O. Calvo and S. Rossi, "Some Issues on the Design of a Low-Cost Autonomous Underwater Vehicle with an Intelligent Dynamic Mission Planner for Pipeline and Cable Tracking", *Underwater Vehicles*, InTech Open Books, Robotics Series, I-Tech Education and Publishing KG, Viena, Austria, ch. 1, pp. 1-18, 2009.



Bruno V. Menna graduated as Electromechanical Engineer at the Engineering Faculty of Olavarría, National Buenos Aires Province Centre University (FIO - UNCPBA), Argentina (2010). He is a Ph.D. student in the same institution, working in the Engineering Group INTELYMEC and with a scholarship of the Argentinean National Research Council (CONICET). With Phd. Thesis related to underwater acoustic communication and positioning systems. bruno.menna@fio.unicen.edu.ar.



Sebastian A. Villar graduated as Systems Engineer at the Exact Sciences Faculty of National Buenos Aires Province Centre University (UNCPBA), Argentina (2009), and as Masters in Business Administration (MBA) Economic Science Faculty of UNCPBA (2011), and as Ph.D. in Engineering, at Engineering Faculty of UNCPBA (2014). He is also a researcher of the Argentinean National Research Council (CONICET) with a scholarship, working in Engineering Group INTELYMEC. Main topic of interest in SONAR image processing (Av. del Valle 5737-B7400JWI Olavarría; Argentina), UNCPBA. svillar@fio.unicen.edu.ar.



Gerardo G. Acosta graduated as Engineer in Electronics at the National University of La Plata, Argentina (1988), and as Ph.D. in Computer Science, at the University of Valladolid, Spain (1995). He is currently a Full Professor in Control Systems (Electronic Area) and director of the PhD program at the Engineering Faculty at the National Buenos Aires Province Centre University (UNCPBA), Argentina also served as professor of Ph.D. programs in Argentina and Spain. He is also a researcher of the Argentinean National Research Council (CONICET), since 1997 and Director of the Research & Development Group "INTELYMEC", CIFICEN CONICET-UNCPBA. His working interests comprise the use of intelligent control techniques in terrestrial and marine robotics. He has more than one hundred and fifty publications and two copyrights in this and related fields and served as reviewer and scientific committee member of several journals and conferences. Senior Member of the IEEE since 2001, Chairman of the IEEE Computational Intelligence Society Argentinean Chapter (2007-2008), receiving the 2010 Outstanding Chapter Award from CIS, and current Chairman of the IEEE Oceanic Engineering Society Argentinean Chapter, being one of its founders. Member of the Administrative Committee of IEEE OES (2015-2016; 2018-2020). gerardo.acosta@iecc.org.