

DESARROLLO Y VALIDACIÓN DE ELECTRÓNICA MODULAR PARA EL CONTROL Y LA NAVEGACIÓN DE UN ROBOT ACUÁTICO DE SUPERFICIE

Juan Luis Rosendo; Gustavo Schmidt; Fabricio Garelli

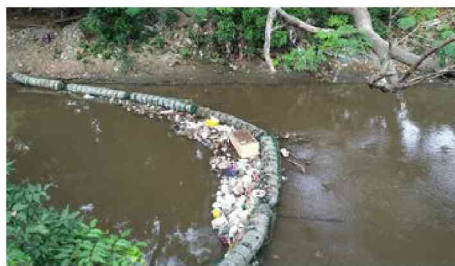
Grupo de Control Aplicado, Instituto LEICI, Dto. de Electrotecnia, Fac. de Ing., UNLP,
Calle 48 esq. 116 s/n, La Plata, Buenos Aires.
juanluis.rosendo@ing.unlp.edu.ar

1. INTRODUCCIÓN

El problema de la contaminación en los cuerpos de agua afecta directamente todas las actividades humanas desde la agricultura, ganadería y pesca, hasta las actividades recreativas [1]. Gran parte de esta contaminación se produce en tierra, en especial cerca de centros urbanos, y luego es arrastrada a través de cursos de agua hacia el océano [2]. Es por esto que en los últimos años ha tomado un marcado interés en la comunidad global su cuidado y recuperación. En este marco pueden encontrarse diversas técnicas de recuperación de desechos y estrategias de control medioambientales [3, 4]. Podemos dividir estas acciones en predictivas, cuyos esfuerzos se centran en evitar que los contaminantes lleguen a los cursos de agua, y las correctivas, que implican la recolección de los residuos desde los flujos de agua.

Dentro de las técnicas correctivas, las técnicas de filtrado constituyen una de las más simples de implementar. Éstas consisten en la recolección temprana de los contaminantes cuando aún se encuentran en flotación, ya que la recuperación de los mismos una vez alcanzado el fondo incurre en mayores costos e incluso pueden dañar el ecosistema formado alrededor de ellos. Este tipo de solución tradicionalmente implica la colocación de barreras colectoras de residuos que luego son vaciadas manualmente, o la utilización de barcazas recolectoras con el mismo fin [4]. Ambas soluciones constituyen tareas automatizables, pero que en el enfoque tradicional requieren al menos la operación por parte de un operario que puede estar expuesto a contaminantes perjudiciales para su salud [5], ver Fig. 1.

En este contexto es donde se encuadra la solución propuesta en este trabajo: un robot autónomo que realice un barrido sistemático recolectando residuos superficiales y a su vez realice un relevamiento hidroambiental generando un mapa de control de calidad de agua. Este desarrollo es producto de un acuerdo entre una empresa franco-argentina, Recyclamer Inovation [6],



(a) Barrera recolectora de basura



(b) Recolección manual de residuos

Figura 1: Enfoques tradicionales para la recuperación de residuos en flotación.

y el Instituto LEICI (FI-UNLP-CONICET), materializado a través de un servicio tecnológico de alto nivel (STAN 5521) [7]. Producto de este acuerdo la empresa aporta su conocimiento y desarrollos en tecnologías para la limpieza de cursos de agua, mientras que desde el LEICI se aporta con el dimensionado y montaje de la electrónica de control, junto a los algoritmos de navegación y operación de la plataforma.

2. DESARROLLO EXPERIMENTAL

Con la premisa de un diseño modular, todo el equipo fue desarrollado considerando el testeo individual de cada parte, para luego integrarse a un conjunto. A continuación se describen las estructuras de hardware y software utilizadas. En primer lugar se presenta y detalla el hardware utilizado ya que el mismo limita las alternativas de software.

2.1. Estructura de hardware

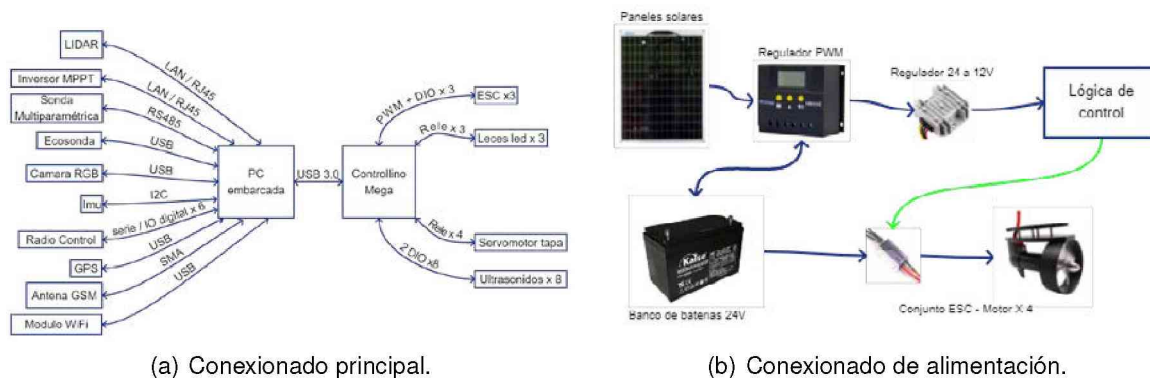


Figura 2: Estructura de hardware

La estructura del robot se basa en un casco de tipo catamarán con la parte posterior cerrada, ver Fig 3(c). La misma cuenta con tres motores de tipo brushless para el comando de dirección y un cuarto motor interno que se utiliza para generar el efecto de succión a través de la boca frontal de la plataforma. La parte inferior de esta boca, durante la operación normal, queda sumergida unos 30 centímetros lo que permite el ingreso de elementos en flotación en las cercanías del robot.

La estructura de conexionado de hardware se muestra esquemáticamente en la Fig. 2(a). De la misma puede observarse que se basa en la interconexión de dos elementos de control. Por una lado una PC embarcada que se encarga de todo el procesamiento de alto nivel, así como la conexión de sensores “inteligentes”, y por otro una plataforma Controllino Mega que se encarga del comando de los variadores de velocidad (ESC - Electronic Speed Controller) utilizados en los motores, así como de otros sensores de bajo nivel. La justificación de una estructura de procesamiento separada, radica en brindar una dedicación plena por parte del control secundario a la atención del control de motores y a los ocho sensores de tipo ultrasónicos utilizados por la plataforma en las tareas de evitación de obstáculos. Desde el punto de vista de conexionado y posibles agregados futuros, la utilización de una base similar a un PLC permite la interconexión de diversos sensores de bajo nivel (con entradas/salidas digitales o analógicas).

Finalmente el montaje de la electrónica se realiza en dos cajas estancas, que se ubican en la parte inferior de la tapa del robot. La elección de esta ubicación obedece a que se deseaba un fácil acceso y montaje. En caso de reparaciones un técnico podría simplemente desconectar los conectores eléctricos de las cajas, soltar los agarres y reemplazarlas, facilitando la repa-

ración. La Fig. 3 muestra una distribución parcial de componentes en las cajas estancas. La elección de utilizar dos cajas estancas obedeció a una cuestión de espacio en la plataforma, y la distribución de componentes en las mismas se vio influenciada por sus aplicaciones. Logrando una caja dedicada a la etapa de potencia donde se encuentran los ESCs y el regulador de carga del panel solar (Fig. 3(b)), y por otro lado una caja con la PC y el control secundario junto a los sensores y/o conexiones hacia ellos (Fig. 3(a)).

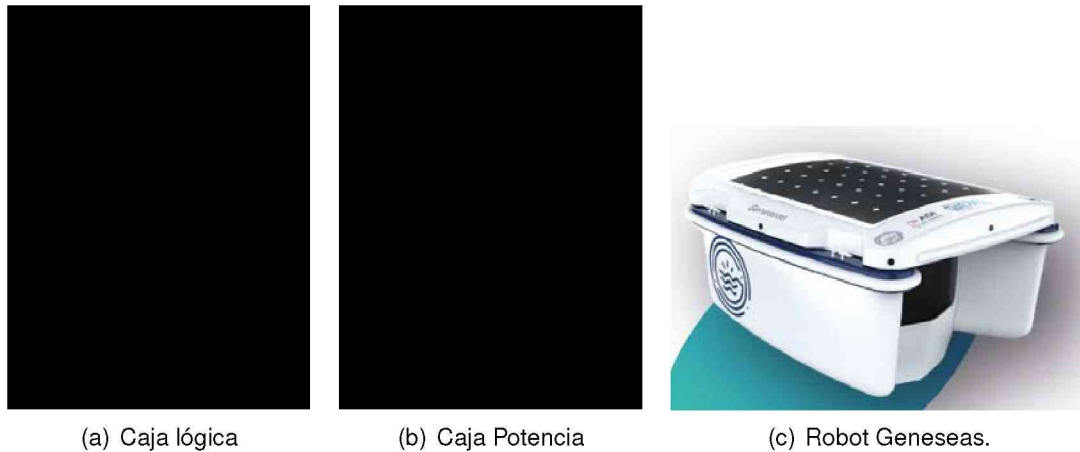


Figura 3: Hardware implementado

Respecto a la interconexión de los sensores, si bien una solución de tipo bus (por ejemplo utilizando tecnologías como CANBUS o NMEA, propia de sistemas náuticos) hubiese gestionado una estructura más simple, la premisa del desarrollo fue basarse en sensores ya utilizados por la empresa contratante, en lo posible económicos y de rápido acceso en el mercado. Esto ha sido a costa de incorporar varios módulos adaptadores para su conexión con la PC embarcada. En la Fig. 2(a) es posible observar la variedad de interfaces utilizadas en el diseño. Se remarca que la posibilidad de interconectar sensores con diferentes interfaces y protocolos fue uno de las premisas del desarrollo.

Con respecto a la alimentación del sistema, la misma se realiza a partir de dos baterías de 12V 100AH AGM de ciclo profundo puestas en serie formando un banco de 24V. La elección de esta tensión de alimentación es debido al tipo de motores disponibles en el robot. Adicionalmente el robot montado en el LEICI cuenta con dos paneles solares de 50W que recargan parcialmente las baterías durante el funcionamiento. Considerando un funcionamiento continuo luego de una recarga completa y con un día soleado, la autonomía del dispositivo es de 6 horas de trabajo. La Fig. 2(b) muestra esquemáticamente la conexión de alimentación utilizada.

2.2. Estructura de software

El software de control se basa en el sistema ROS (Robot Operating System) [8]. La elección de este sistema obedece a dos razones: la propia modularidad del mismo, y la amplia cantidad de código libre disponible para dar soporte a desarrollos. ROS se trata de una serie de frameworks, librerías y paquetes que permite la comunicación entre distintos procesos (nodos en la jerga de ROS) y su control de ejecución.

La Fig. 4(a) muestra esquemáticamente los principales nodos que intervienen en el algoritmo de control. La estructura de software se monta a partir de un nodo principal el cual es iniciado en el arranque del robot. Su función principal es la de chequear las configuraciones y estados iniciales del robot (valores de medida de sensores, ubicación respecto a lo esperado por la aplicación de control, estado de baterías, etc.). Una vez concluida esta etapa de chequeo, este

mismo nodo activa los nodos de control manual o control automático. Esta activación es función del estado de un interruptor en el control de radio frecuencia del robot, el cual es informado mediante el nodo serie indicado en la figura antes dicha. Es importante notar que en el gráfico anterior todos los nodos se ejecutan en la PC embarcada con excepción del nodo serie que se ejecuta sobre el control secundario (Controllino). Es este nodo serie el que será el encargado de realimentar al sistema principal las mediciones de los sensores de bajo nivel (entre ellas la lectura de los sensores ultrasónicos) y de actualizar los comandos a los motores generados por los nodos principales. La comunicación entre el Controllino y la PC embarcada se realiza mediante un protocolo USB, sobre el que se establece una comunicación serie que permite al sistema ROS alcanzar el nodo desarrollado en el control secundario. Una vez seleccionado el modo de funcionamiento, el comportamiento del robot queda regido por los nodos manual o automático, en particular el control manual solo retransmite y mapea las señales recibidas del control por radiofrecuencia al comando de motores.

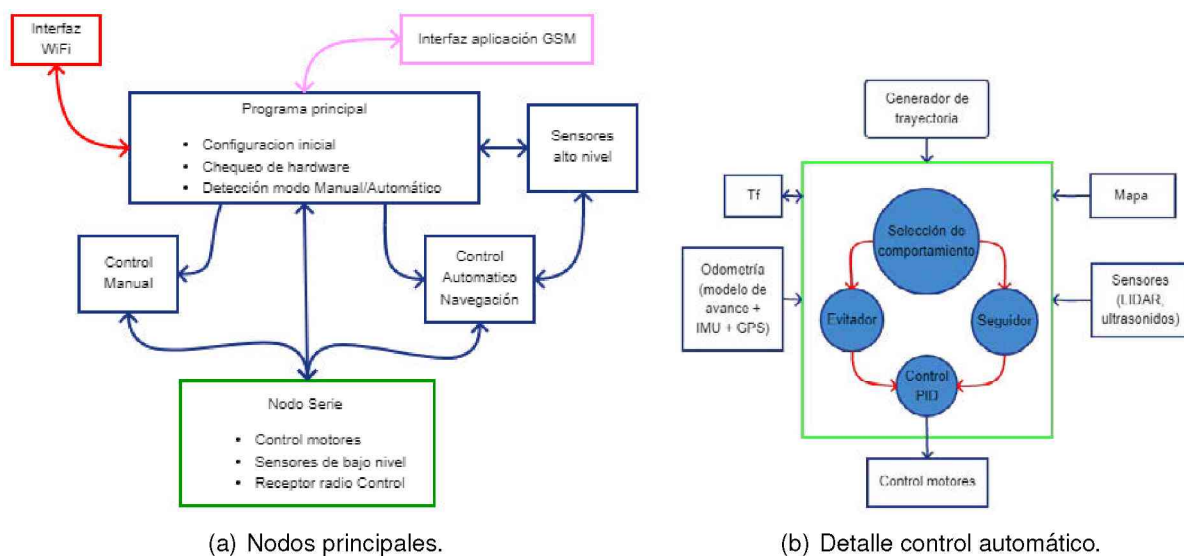


Figura 4: Estructura de software

Por otra parte, el control automático es más complejo e implica varios elementos trabajando en forma conjunta, ver Fig. 4(b). El sistema parte de generar un mapa del área de trabajo en el formato de grilla de ocupación a partir de dos coordenadas GPS obtenidas de la aplicación web. El objetivo de este mapa es almacenar información con un cierta probabilidad de que las celdas estén ocupadas o no. Un nodo independiente genera trayectorias aleatorias de barrido a ser seguidas por el robot. Estas trayectorias son ejecutadas por el robot en forma inmediata, pero a cada paso se chequea el estado de los sensores de proximidad. En caso de no detectar obstáculos en un área de peligro dada (espaciamiento de 1m alrededor del robot), se ejecuta el comando de trayectoria a través de un nodo seguidor que implementa un control PID que corrige el comando de velocidad y dirección. En caso de detectar un obstáculo, se ingresa en un comportamiento diferente del robot que modifica la referencia a seguir buscando rodear al objeto hasta lograr superarlo (algoritmo BUG2 [9]) y continuar la trayectoria generada. Si el robot pasa más de un cierto tiempo (parámetro ajustable) sin poder liberar el obstáculo, se genera una nueva trayectoria en forma aleatoria. Este último comportamiento busca liberar al robot de posibles situaciones de trampa donde el algoritmo de evitación pueda fallar o ante cambios en el área de trabajo.

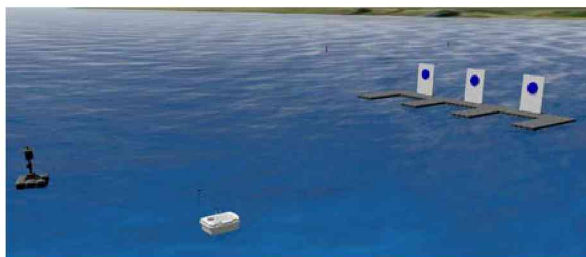
Para ubicar al robot en el mapa se implementa un filtro de Kalman [10], el cual utiliza información proveniente de un modelo del robot y calcula su odometría a partir de los comandos de velocidad enviados a los motores y las lecturas del sensor IMU y GPS. Todos estos elementos

presentes en el robot tienen posiciones fijas que modifican sus lecturas respecto del marco del robot. Para el mapeo de los datos recibidos en el marco del robot se utiliza el paquete TF perteneciente al sistema ROS [11].

3. RESULTADOS Y DISCUSIÓN

Como primer paso hacia la realización de ensayos se adaptó el simulador Virtual RobotX (VRX) [12], para visualizar comportamientos del robot en desarrollo. Se modificó el modelo original del robot WAM-V brindado en el simulador [13], y se reemplazaron las características físicas y visuales del robot Geneseas. El modelado del robot se basó en las ideas presentadas por Foseen [14], y los parámetros físicos a utilizar se estimaron a partir de considerar al robot en desarrollo como volúmenes simples (masa e inercia). El trabajo en el simulador permitió adelantarse a pruebas de campo, en particular a operaciones de navegación como seguimientos de puntos GPS, evitación de obstáculos y reconocimiento de imágenes para el atraque autónomo. En la Fig. 5(a) se puede ver una imagen del simulador.

Los primeros ensayos experimentales consistieron en chequear el funcionamiento individual de cada componente del sistema, y la respuesta del control manual. Estos fueron realizados en las piletas del Departamento de Hidráulica de la Facultad de Ingeniería de la UNLP en diciembre de 2022, ver Fig. 5(b). En estos ensayos se validaron el modelado realizado en el simulador, las respuestas dinámicas del sistema y el funcionamiento de los sensores embarcados. Asimismo, se realizaron pruebas simples de la conducción autónoma. Los resultados parciales de estos ensayos demostraron la viabilidad de la estructura propuesta tanto a nivel hardware como software, y pusieron en evidencia dificultades a ser trabajadas. Puntualmente, la utilización de sensores ultrasónicos para la evitación de obstáculos en situación de respuestas rápidas, y el ruido que estos introducen al intentar realizar un mapeo a partir de sus lecturas.



(a) Captura del entorno virtual del simulador.



(b) Ensayos realizados en el Depto. de Hidráulica, FI-UNLP.

Figura 5: Ensayos y simulación

4. CONCLUSIONES

Este trabajo presentó el desarrollo parcial del robot Geneseas, justificando las decisiones adoptadas en su construcción y los criterios de ingeniería seguidos para su implementación, como así también detalles constructivos y resultados experimentales preliminares. Al momento del desarrollo de este documento, el equipo de trabajo continúa desarrollando pruebas del funcionamiento autónomo y trabajando en la incorporación de nuevos sensores que optimicen las tareas de barrido y mapeo del entorno. Como principales tareas a desarrollar en el mediano plazo se encuentran la integración de un sensor de tipo LIDAR en reemplazo de los sensores ultrasónicos, la utilización de redes neuronales para el reconocimiento de basura en flotación con el objetivo de optimizar los recorridos, y la implementación de un sistema de atraque autónomo (por el momento solo verificado en simulación).

AGRADECIMIENTOS

Este trabajo fue parcialmente financiado por la Universidad Nacional de La Plata (I253), CONICET (PIP2595) y Recyclamer Inovation. Se agradece a los responsables de Recyclamer Inovation Alan Peral-García y Alejandro Scaglia por la confianza depositada en el grupo de trabajo.

REFERENCIAS

- [1] W. C. Li, H. F. Tse, and L. Fok. Plastic waste in the marine environment: A review of sources, occurrence and effects. *The Science of the total environment*, 566-567:333–349, Oct 2016.
- [2] J. R. Jambeck, R. Geyer, C. Wilcox, T. R. Siegler, M. Perryman, A. Andrady, R. Narayan, and K. L. Law. Plastic waste inputs from land into the ocean, 2015.
- [3] Ronald L. Gehr Ronald L. Droste. *Theory and Practice of Water and Wastewater Treatment*. September 2018.
- [4] Olivia K. Helinski, Cara J. Poor, and Jordyn M. Wolfand. Ridding our rivers of plastic: A framework for plastic pollution capture device selection. *Marine Pollution Bulletin*, 165:112095, 2021.
- [5] Bashir Yahaya Sanda and Isyaku Ibrahim. Causes categories and control of water pollution. *International Journal of Scientific Engineering and Science*, 2020.
- [6] Recyclamer innovation. <https://www.recyclamer-innovation.com/>.
- [7] Fabricio Garelli and Juan Luis Rosendo. Asesoramiento técnico para la navegación para vehículos marinos autónomos de superficie. *STAN5521 CONICET*, 2022.
- [8] Morgan Quigley, Brian Gerkey, Ken Conley, Josh Faust and Tully Foote, Jeremy Leibs, Eric Berger, Rob Wheeler, and Andrew Ng. Ros: an open-source robot operating system. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA) Workshop on Open Source Robotics*, Kobe, Japan, May 2009.
- [9] Seth Hutchinson George Kantor Wolfram Burgard Lydia E. Kavraki Sebastian Thrun Howie Choset, Kevin M. Lynch. *Principles of Robot Motion: Theory, Algorithms, and Implementations*. Intelligent Robotics and Autonomous Agents. The MIT Press, 2005.
- [10] Peter Corke. *Robotics, Vision and Control: Fundamental Algorithms in MATLAB*. Springer Tracts in Advanced Robotics 73. 2011.
- [11] Wim Meeussen Tully Foote, Eitan Marder-Eppstein. tf2. <http://wiki.ros.org/tf2>.
- [12] Brian Bingham, Carlos Agüero, Michael McCarrin, Joseph Klamo, Joshua Malia, Kevin Allen, Tyler Lum, Marshall Rawson, and Rumman Waqar. Toward maritime robotic simulation in gazebo. In *Proceedings of MTS/IEEE OCEANS Conference*, Seattle, WA, October 2019.
- [13] Edoardo Sarda, Huajin Qu, Ivan Bertaska, and Karl von Ellenrieder. Station-keeping control of an unmanned surface vehicle exposed to current and wind disturbances. *Ocean Engineering*, 127:305–324, November 2016.
- [14] Thor I. Fossen. *Handbook of Marine Craft Hydrodynamics and Motion Control*. Wiley, 2011.