

Fusión de mapas mediante encuentros frecuentes para la exploración multirobot

Martin Nievas*, Gastón Araguás[†] y Claudio J. Paz[‡]

Centro de Investigación en Informática para la Ingeniería,
Universidad Tecnológica Nacional, Facultad Regional Córdoba
Email: {*,[†]garaguas, [‡]cpaz}@frc.utn.edu.ar

Resumen—Las estrategias de exploración son cruciales para que los equipos de robots puedan explorar y navegar de manera eficiente un ambiente desconocido. En este trabajo se presenta una estrategia de exploración para coordinar la exploración de un equipo de robots con encuentros frecuentes y comunicación limitada. El enfoque propuesto aprovecha los encuentros entre robots para intercambiar informaciones y generar un mapa global a través de la fusión de los mapas parciales de cada robot, mediante un esquema de dos etapas utilizando los algoritmos SVD e ICP. Se presenta un esquema para la detección de fronteras tratando los mapas generados por los robots como imágenes, y guiar la exploración. Finalmente, se presentan resultados tanto de simulación como experimentación en el mundo real.

Keywords—multirobot, simulación, exploración, ROS2

I. INTRODUCCIÓN

Construir mapas es una tarea fundamental en la robótica móvil, ya que permite a los robots conocer el entorno en el que se encuentran y planificar rutas seguras. El concepto de exploración puede definirse como el control de un robot móvil en un ambiente inicialmente desconocido, mientras genera un mapa del mismo que puede ser utilizado posteriormente para la navegación. Existen numerosas situaciones en las que un grupo de robots explora un ambiente desconocido, entre las cuales se pueden mencionar aplicaciones de transporte [1], industria [2], [3], agricultura [4]–[6], entre otras. En estos ambientes, los robots tienen que poder navegar sin intervención humana.

Una de las tareas asignadas al grupo de robots es la coordinación para realizar de forma eficiente la exploración. Para esto, los robots se dirigen a lugares desconocidos (o fronteras), pero que no fueron explorados por ninguno de los miembros del equipo. Explorar un ambiente desconocido puede ser complejo o simple, dependiendo de la cantidad y distribución de los obstáculos en el ambiente. En ambos casos, el equipo de robots parte sin información del entorno y tienen que construir una representación de forma progresiva a partir de las mediciones de sus sensores. Los robots inician la operación de exploración sin conocer la posición de los obstáculos en el mapa, y en ocasiones sin conocer la posición inicial de los miembros del equipo.

En escenarios de exploración con gran cantidad de obstáculos, las comunicaciones pueden tener un alto grado de incertidumbre comprometiendo la confiabilidad, capacidad, y la disponibilidad de enlaces entre los nodos en una red [7]. Si bien un robot puede explorar de forma autónoma mientras está fuera del alcance de comunicación con la base por un tiempo,

la solución tiene que contemplar que el área explorada por un robot sea comunicada al equipo para integrarla al mapa global.

En este trabajo se presenta un sistema de coordinación de exploración, para un grupo de robots que opera en un ambiente no estructurado con encuentros frecuentes y comunicación limitada. El enfoque propuesto utiliza los encuentros entre robots para intercambiar información y generar un mapa global mediante la fusión de los mapas parciales de cada robot, mediante un esquema de dos etapas.

El trabajo está organizado de la siguiente manera: en la Sección II se realiza un repaso por las técnicas relacionadas a la exploración con múltiples robots. La Sección III presenta la formulación del problema y en la Sección IV el método propuesto. Finalmente en la Sección VI se mencionan los resultados obtenidos, las limitaciones y posibles trabajos futuros.

II. TRABAJOS RELACIONADOS

Existen diferentes estrategias para la exploración de entornos desconocidos, entre las cuales se pueden mencionar la exploración a la frontera más próxima. El concepto de frontera introducido en [8] implica que una celda es conocida y las contiguas no. Esta técnica proporciona el camino más corto para un solo robot [9]. Sin embargo, este método no tiene en cuenta la cantidad de información que se va a generar en un determinado punto de observación, solo determina el camino más próximo a un lugar no explorado.

Otra técnica es la exploración basada en ganancia de información, la cual se basa en la teoría de la información para determinar la mejor posición para adquirir una nueva medición. Solo se tiene en cuenta la cantidad de información que se incorpora desde un cierto punto de vista en el mapa. Como no es posible conocer a priori qué medición se leerá en esa posición, se debe integrar todas las medidas posibles para calcular la ganancia de información en ese punto. En [10] utilizan un proceso de raycasting para calcular la ganancia de información en cada frontera. Este proceso es lento, ya que implica realizar este proceso con diferentes orientaciones del robot en múltiples puntos del mapa.

En [11] presentan una estrategia de exploración que utiliza un árbol de exploración RRT para guiar la exploración, mientras se determina la ganancia de información. Se generan posibles trayectorias sobre el espacio libre utilizando el algoritmo RRT, y solo el primer tramo de la mejor trayectoria es ejecutado.

Contar con múltiples robots para la exploración permite reducir el tiempo de exploración y aumentar la cobertura del mapa [12]. Si los robots pueden compartir información entre ellos, pueden ser coordinados para explorar de forma más eficiente. Por otro lado, la presencia de múltiples robots implica que se deben coordinar para evitar colisiones y superposición de tareas. En [12] presentan una técnica de exploración para un equipo de robots con sensores de distancia. Se construye un mapa del área explorada junto a las zonas seguras, en forma de grafo. Utilizan algoritmos descentralizados para asegurar la eficiencia de la exploración y evitar conflictos. En [13] proponen un algoritmo para la exploración de un entorno desconocido por un equipo de robots. La tarea de exploración es dividida en dos etapas: identificar los posibles objetivos para los robots y asignar a cada robot un objetivo identificado anteriormente.

En un contexto multi robot, la asignación de objetivos es una tarea importante, ya que se debe minimizar el trabajo redundante y la interferencia entre los robots. En [13] se propone una técnica de segmentación del mapa en regiones, donde cada región es asignada a un robot para su exploración. La segmentación utilizada está relacionada con la estructura semántica introducida en [14], mientras que la asignación de objetivos se basa en el método Húngaro [15]. En [16] introducen el concepto de Gap Navigation Tree, en la cual suponen que los obstáculos son únicamente diferenciables entre ellos. Sin embargo, esto no siempre es posible de lograr, ya que en ambientes altamente simétricos se pueden generar falsas asignaciones.

III. FORMULACIÓN DEL PROBLEMA

El entorno para explorar se define como un espacio finito en el cual se encuentran múltiples robots. Se supone que la flota de robots es homogénea y cada uno de ellos está equipado con sensores que le permiten construir un mapa local y localizarse en él utilizando un algoritmo de localización y mapeo simultáneo [17].

Se considera que todos los robots próximos pueden comunicarse entre sí, identificando los robots cercanos. El mapa producido está representado por una grilla de ocupación en la que se identifican celdas fronteras a explorar. Cada robot cuenta con un planificador global y local, el cual utilizando el mapa construido, puede planificar rutas libres de colisión hacia objetivos previamente definidos. La localización inicial de los robots es desconocida y cada robot cuenta con su propio sistema de coordenadas local.

En este documento se propone un método de para calcular la pose entre dos robots que se encuentran próximos entre sí, la fusión de los mapas construidos por cada uno hasta el momento y la coordinación de fronteras a explorar por el conjunto de robots.

La distribución de robots sobre las fronteras puede considerarse como un problema de optimización donde el número de posibles tareas es igual al número de permutaciones sin repetición. Dado que los robots inicialmente no conocen la

posiciones de los otros robots, el problema inicial es considerado exploración de fronteras más cercanas introducido en [8].

Durante el proceso de estimación de pose, debido a que los robots están equipados con sensores Lidar 2D, y al encontrarse próximos el uno del otro, sus mediciones tendrán puntos en común. Para realizar el proceso de estimación de pose entre robots se utiliza un algoritmo llamado ICP (Iterative Closest Point). Este algoritmo se utiliza para alinear dos conjuntos de puntos en un espacio tridimensional. Su objetivo principal es encontrar la mejor transformación, que minimiza la distancia entre los puntos leídos por los sensores láser de los robots. El algoritmo funciona de manera iterativa, comenzando con una estimación inicial de la transformación y luego refinándola sucesivamente hasta converger en una solución óptima o aceptable. Esta transformación inicial es importante, debido a que una mala inicialización puede converger a mínimos locales alejados de la solución óptima. Para solucionar esto, se plantea un esquema de dos etapas en la cual primero se obtiene una estimación de la transformación mediante una descomposición de valores singulares (SVD) de cada medición. Este valor es utilizado como inicialización para el algoritmo ICP

El proceso de exploración consiste en llevar a cada robot a la frontera más cercana que no haya sido explorada por otro robot. Dado un mapa de ocupación, se define como frontera como una celda que tiene al menos un vecino no explorado. Este concepto fue introducido en [8] y es utilizado en la planificación de tareas para robots móviles.

IV. DESCRIPCIÓN DEL SISTEMA

IV-A. Extracción de fronteras

Partiendo del mapa de ocupación, la misma se convierte en una imagen en escala de grises como se puede observar en la Figura 1a. De esta imagen se extraen los obstáculos correspondientes a las celdas ocupadas, para luego aplicar una máscara de dilatación y unir las celdas adyacentes que puedan haber quedado desconectadas debido a la dispersión del láser. Con esto se consigue una imagen binaria que representa los bordes de los obstáculos representada en la Figura 1b. Luego, se genera una máscara a partir de las celdas libres y se realiza un proceso de dilatación sobre la imagen para obtener una máscara de celdas exploradas (obstáculos y libres). Este proceso se aplica para reducir el ruido en la imagen generado por el algoritmo de SLAM, debido al proceso de raytracing en el mapa. Teniendo la imagen de las celdas exploradas (obstáculos y libres) y de las celdas ocupadas, se aplica una operación de resta entre las mismas para obtener los límites entre las zonas exploradas y no exploradas. Estas corresponden a las fronteras a explorar, representados por los píxeles blancos en la Figura 1c.

Partiendo de los contornos de la imagen, se calculan los centroides de las celdas no exploradas, los cuales representan los puntos de interés a explorar. En la Figura 1d, se pueden observar los contornos frontera en verde y las coordenadas del centroide p_f como puntos rojos.

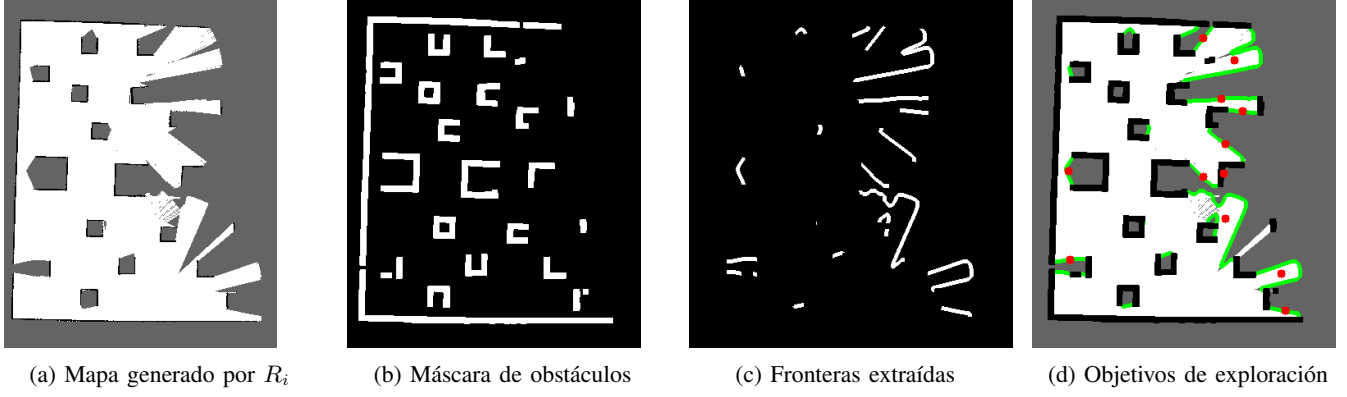


Figura 1: (a) Mapa de ocupación generado por el robot durante el proceso de exploración. (b) Imagen binaria de celdas con obstáculos, en blanco las celdas ocupadas. (c) Imagen binaria de fronteras detectadas, en blanco zonas de fronteras. (d) Mapa de ocupación con fronteras en color verde y coordenadas utilizadas para guiar la exploración en rojo.

Como los puntos p_f están en el sistema de coordenadas de la imagen en pixel, se los convierte al sistema de coordenadas del mapa utilizando la resolución del mapa y el origen del mismo.

Como resultado final se obtiene una lista de puntos:

$$L_f = [f_1, f_2, \dots, f_n] \in {}^M f_i = \{x, y\} \quad (1)$$

en el sistema de coordenadas del mapa, que representan las fronteras a explorar.

IV-B. Coordinación de robots

Considerando ${}^M \xi_{Bi}$ como la transformación del robot R_i (Body) en el mundo, y ${}^W \xi_{Mi}$ la transformación del mapa en el robot R_i , se define la transformación del mapa al mundo como:

$${}^W \xi_{Bi} = {}^W \xi_{Mi} \cdot {}^M \xi_{Bi} \quad (2)$$

Un robot, al detectar que otro se encuentra cerca, procede a calcular la transformación entre los mismos. Para la coordinación de los robots durante el encuentro se utiliza una máquina de estados, la cual se encarga de coordinar la fusión de los mapas y la selección de las fronteras a explorar. El estado normal de un robot es *exploración*, en el cual el robot se mueve hacia las fronteras a explorar.

Cuando un robot detecta a otro robot cerca, se pasa al estado *encuentro*, en el cual se determina el robot que se encargará de fusionar los mapas y se calcula la transformación entre los robots. Cada robot cuenta con un *ID* único, el cual es utilizado para identificarse frente a los demás robots. El robot que se encarga de fusionar los mapas es el que tiene el *ID* más bajo y se encarga de calcular la transformación entre los robots. Cuando un robot detecta que tiene el menor *ID* entre los robots cercanos, se encarga de calcular la transformación al otro robot y pasa al estado *encuentro confirmado*. Para esto, el robot se suscribe el tópico *scan* del otro robot y calcula la transformación entre los robots en un proceso de dos etapas.

En la primera etapa, se utiliza la técnica de descomposición de valores singulares (SVD) para estimar la transformación entre los robots. Dadas las mediciones láser L_1 y L_2 de los robots R_1 y R_2 respectivamente, se calcula el centroide de las mediciones C_1 y C_2 , para obtener la matriz de covarianza H_{12} entre los robots como:

$$H_{12} = C_1^T \cdot C_2. \quad (3)$$

Luego, se descompone la matriz H_{12} en valores singulares $U \Sigma V^T$ y se calcula la rotación R y traslación T entre los robots como:

$$R = VU^T, \quad (4)$$

y

$$T = C_2 - R \cdot C_1. \quad (5)$$

Con esto se obtiene una estimación de la transformación ${}^{B_1} \xi_{B_2}$ entre la base del robot R_1 y la base del robot R_2 .

Dado que la estimación de la transformación es una aproximación, se utiliza el algoritmo de ICP para refinar esta última. Para esto se transforma la medición láser L_2 mediante T y R calculada recientemente y se aplica el algoritmo de ICP para obtener la transformación refinada ${}^{B_2} \xi_{B_1}^{ref}$.

Una vez finalizado el proceso de cálculo de la transformación, el robot que se encarga de fusionar los mapas se suscribe al tópico *world_pose* del otro robot y cambia al estado *esperar pose* para obtener la transformación ${}^W \xi_{B_2}$.

Una vez recibida la transformación ${}^W \xi_{B_2}$, se calcula la transformación ${}^W \xi_{M_1}$ de la forma:

$${}^W \xi_{M_1} = {}^W \xi_{B_2} \cdot {}^{B_2} \xi_{B_1}^{ref} \cdot {}^{M_1} \xi_{B_1}^{-1}. \quad (6)$$

IV-C. Fusión de mapas

Una vez calculada la transformación entre los robots, se procede a fusionar los mapas. Para esto, el robot encargado de fusionar los mapas R_1 se suscribe al tópico *map* del otro robot y obtiene el mapa construido por el mismo. Como ${}^W \xi_{M_1}$

y $W_{\xi_{M2}}$ son valores obtenidos previamente, se puede calcular la transformación entre los mapas como:

$$M_2 \xi_{M1} = W_{\xi_{M2}}^{-1} \cdot W_{\xi_{M1}}. \quad (7)$$

Previo a la fusión de mapas, se calculan los puntos extremos de los mapas y se los transforma al sistema de coordenadas del mapa de R_1 . Con estos límites se crea un mapa vacío donde todas sus celdas son desconocidas. En la Figuras 2a y 2b se pueden observar los mapas construidos por el robot R_1 y R_2 respectivamente, y el mapa fusionado resultante en la Figura 3. Una vez actualizado el mapa, se procede a actualizar las fronteras a explorar, descartando las fronteras que ya fueron exploradas por el otro robot. Para esto, se repite el proceso de detección de fronteras. Debido a que el mapa fusionado tiene nuevas celdas libres, las fronteras previamente detectadas en esta zona son eliminadas por el proceso de enmascaramiento detallado previamente.

V. EXPERIMENTACIÓN

V-A. Simulación

Para la experimentación se utilizó el simulador Gazebo, el cual permite simular robots móviles y sensores, y el paquete de ROS 2 *nav2_slam* para la localización y mapeo simultáneo. Los robots utilizados son dos robots con tracción diferencial, los cuales cuentan con un sensor láser 2D. En ambiente a explorar está compuesto por un galpón de 60x40 metros en el cual se encuentran columnas, paredes, estanterías y cajas.

En la Figuras 2a y 2b se pueden observar los mapas contruidos por el robot R_1 y R_2 respectivamente, dentro de la simulación. Cada robot explora una sección diferente del mapa hasta su encuentro. En la Figura 2a se puede apreciar el recorrido realizado por R_1 , junto a las fronteras que quedaron explorar en su mapa local.

V-B. Robot reales

Para la experimentación con robots se utilizó un robot con tracción diferencial, el cual cuenta con un sensor láser 2D de 360 grados. Debido a limitaciones físicas para acceder a dos robots en forma simultanea, se realizó un experimento para validar los resultados con un solo robot sin perder la generalidad para múltiples robots. En una primera instancia, el robot R_1 comienza explorando la parte superior de un pasillo de oficinas hasta llegar a una esquina y encontrarse con el robot R_2 como se puede ver en la Figura 4. Esta secuencia es almacenada en un *rosvag* para ser utilizado posteriormente.

En la segunda etapa, el robot R_2 inicia su exploración en la zona inferior del pasillo, desplazándose hacia la derecha para llegar a un pasillo cerrado.

Luego rota hacia el pasillo de la izquierda para esperar su encuentro con el robot R_1 como se puede ver en la Figura 4. Durante esta última secuencia, mientras R_1 está detenido, se reproduce el archivo *rosvag* grabado anteriormente con las mediciones de R_2 . Este, al llegar al lugar de encuentro, es detectado por R_1 quien realiza el proceso de fusión de mapas. En la Figura 4 se pueden observar los mapas fusionados.

En las Figuras 4a y 4b se pueden observar los mapas contruidos por R_1 y R_2 respectivamente al momento del encuentro en el pasillo 4c y 4d. Mientras que en la Figura 4e se observa la fusión de los mapas realizados por el robot R_1 .

VI. COMENTARIOS FINALES

En este trabajo se presentó un algoritmo para coordinar y fusionar mapas generados por un equipo de robots que exploran un ambiente no estructurado. También se presentó un algoritmo de dos etapas para estimar la pose entre los robots, considerando que las mediciones láser tienen puntos en común por la cercanía entre los robots. En la primera etapa se utiliza la descomposición mediante SVD para utilizar estos valores como inicialización de un algoritmo de ICP.

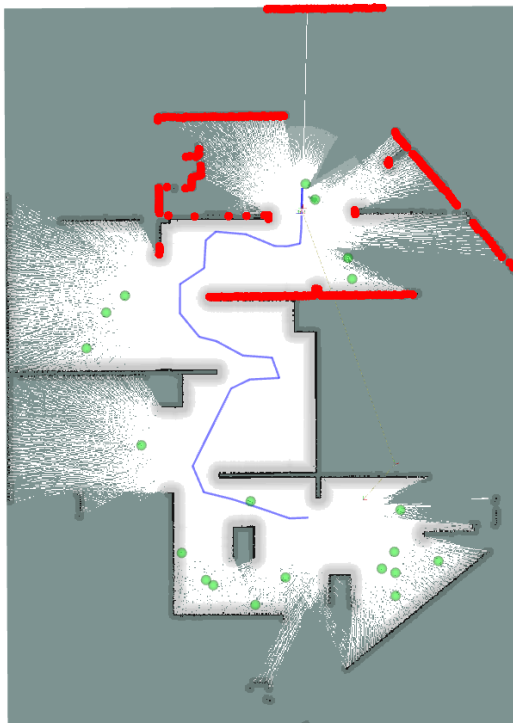
Se realizaron pruebas en simulación para validar los algoritmos presentados de forma exhaustiva, obteniendo un mapa del ambiente a partir de los mapas fusionados de los robots. También se realizaron pruebas con un robot real y datos grabados, logrando un mapa fusionado.

La implementación actual del algoritmo de ICP está realizada en CPU y no se obtiene un puntaje sobre las iteraciones, por lo que todas las transformaciones estimadas se consideran válidas. Trabajos futuros contemplará utilizar este puntaje para descartar las mediciones, y reiniciar el proceso de fusión. También se prevé utilizar un algoritmo de ICP implementado en GPU y realizar la fusión de mapas directamente en la misma.

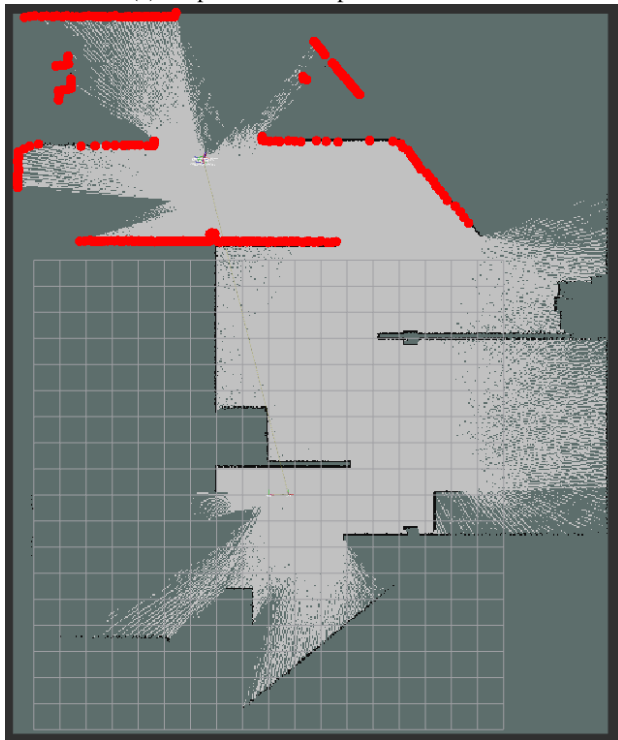
El algoritmo actual contempla que la fusión se realiza cada vez que los robots se encuentran próximos. Si los robots se encuentran en un pasillo, el proceso de fusión se repite por cada paso de exploración generando demoras en la navegación. Esto podría ser solucionado si cada robot contara con una lista con los lugares, el tiempo y el *ID* del robot con el que se encontró, y no volver a fusionar mapas hasta pasado un tiempo. Esto será contemplado en un trabajo futuro.

REFERENCIAS

- [1] J. Bae and M. Park, "A heuristic for efficient coordination of multiple heterogeneous mobile robots considering workload balance," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 4064–4070, 2021.
- [2] M. Manoharan, A. N. Shridhar, V. Y. Vinod, and S. Kumaraguru, "A novel volume decomposition methodology for multi-robots collaborative additive manufacturing," in *2020 IEEE 4th conference on information & communication technology (CICT)*. IEEE, 2020, pp. 1–6.
- [3] M. Saboia, L. Clark, V. Thangavelu, J. A. Edlund, K. Otsu, G. J. Correa, V. S. Varadharajan, A. Santamaria-Navarro, T. Touma, A. Bouman *et al.*, "Achord: Communication-aware multi-robot coordination with intermittent connectivity," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 10 184–10 191, 2022.
- [4] A. Dutta, S. Roy, O. P. Kreidl, and L. Bölöni, "Multi-robot information gathering for precision agriculture: Current state, scope, and challenges," *Ieee Access*, vol. 9, pp. 161 416–161 430, 2021.
- [5] X. Kan, T. C. Thayer, S. Carpin, and K. Karydis, "Task planning on stochastic aisle graphs for precision agriculture," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 3287–3294, 2021.
- [6] T. C. Thayer, S. Vougioukas, K. Goldberg, and S. Carpin, "Multirobot routing algorithms for robots operating in vineyards," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 3, pp. 1184–1194, 2020.



(a) Mapa construido por el robot R_1 .



(b) Mapa construido por el robot R_2 .

Figura 2: Mapas construidos por los robots R_1 y R_2 . En rojo mediciones del sensor Lidar 2D, en verde las coordenadas de las fronteras marcadas para explorar y en violeta el camino realizado por el robot.

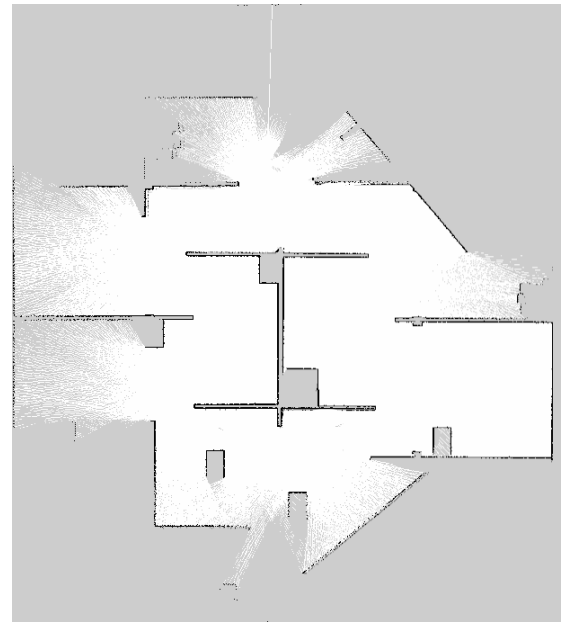
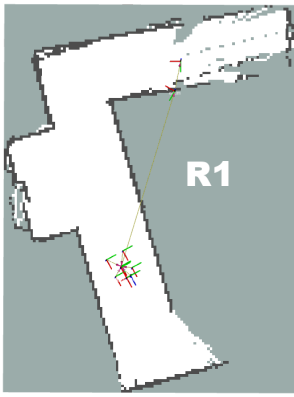
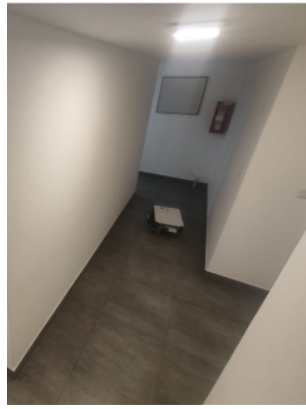


Figura 3: Mapas de R_1 y R_2 fusionados. El mapa presenta errores de desplazamiento, debido a que la transformación estimada no es exacta.

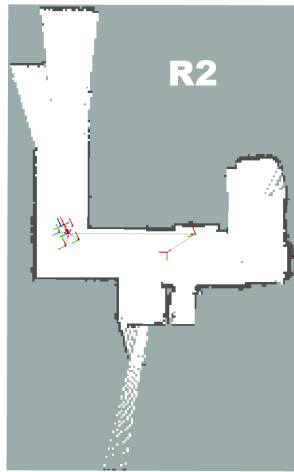
- [7] A. Husain, H. Jones, B. Kannan, U. Wong, T. Pimentel, S. Tang, S. Daftry, S. Huber, and W. L. Whittaker, "Mapping planetary caves with an autonomous, heterogeneous robot team," in *2013 IEEE Aerospace Conference*. IEEE, 2013, pp. 1–13.
- [8] B. Yamauchi, "A frontier-based approach for autonomous exploration," in *Computational Intelligence in Robotics and Automation, 1997. CIRA'97. Proceedings., 1997 IEEE International Symposium on*. IEEE, 1997, pp. 146–151.
- [9] S. Thrun, "Probabilistic robotics," *Communications of the ACM*, vol. 45, no. 3, pp. 52–57, 2002.
- [10] H. H. González-Banos and J.-C. Latombe, "Navigation strategies for exploring indoor environments," *The International Journal of Robotics Research*, vol. 21, no. 10-11, pp. 829–848, 2002.
- [11] A. Bircher, M. Kamel, K. Alexis, H. Oleynikova, and R. Siegwart, "Receding horizon"next-best-view"planner for 3d exploration," in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 1462–1468.
- [12] A. Franchi, L. Freda, G. Oriolo, and M. Vendittelli, "The sensor-based random graph method for cooperative robot exploration," *IEEE/ASME Transactions on Mechatronics*, vol. 14, no. 2, pp. 163–175, 2009.
- [13] K. M. Wurm, C. Stachniss, and W. Burgard, "Coordinated multi-robot exploration using a segmentation of the environment," in *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*. IEEE, 2008, pp. 1160–1165.
- [14] B. Kuipers and Y.-T. Byun, "A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations," *Robotics and autonomous systems*, vol. 8, no. 1-2, pp. 47–63, 1991.
- [15] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval research logistics quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.
- [16] R. Nasir and A. Elnagar, "Gap navigation trees for discovering unknown environments," *Intelligent Control and Automation*, vol. 6, no. 04, p. 229, 2015.
- [17] S. Macenski, F. Martín, R. White, and J. Ginés Clavero, "The marathon 2: A navigation system," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020. [Online]. Available: <https://github.com/ros-planning/navigation2>



(a) Mapa realizado por R_1



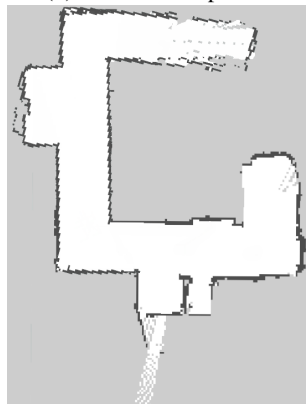
(c) Vista R_1 en pasillo



(b) Mapa realizado por R_2



(d) Vista R_2 en pasillo



(e) Mapa fusionados

Figura 4: (a y c): Mapa generado por R_1 y foto en el pasillo. (b y d): Mapa generado por R_2 hasta su encuentro con R_1 . (e): Mapas fusionados. El mapa presenta errores de desplazamiento y orientación debido que la inicialización de R_1 no está alineado con los ejes cartesianos.