

Robótica de Enjambre

Alberto Zafra Navarro, Antonio García de Mateos Díaz Cacho,
Tamai Ramírez Gordillo y Lluís Bernat Iborra

Enero 2022

Índice

1. Resumen	2
2. Descripción de la tipología en el ámbito de la robótica móvil	2
2.1. Introducción	2
2.1.1. Inteligencia de Enjambre	2
2.1.2. Robótica de Enjambre	3
2.1.3. Propiedades	4
2.1.4. Características	5
2.2. Clasificación	6
2.2.1. Métodos de modelado	6
2.2.2. Tipos de comportamiento colectivo	7
2.3. Aplicaciones	9
2.4. Ejemplos de robots con aplicación en la robótica de enjambre	10
2.4.1. I-Swarm	10
2.4.2. iRobot Swarm	11
2.4.3. Alice Mobile Robot	12
2.4.4. Jasmine Mobile Robot	12
2.4.5. Kilobot project	13
2.4.6. E-Puck Education Robot	13
3. Caso de estudio: Robótica de Enjambre aérea	14
3.1. Introducción	14
3.1.1. Vehículos aéreos no tripulados (UAV)	14
3.1.2. Enjambres aéreos	15
3.2. Descripción teórica	21
3.2.1. Clase <i>Drone</i>	21
3.2.2. Clase <i>Swarm</i>	22
3.2.3. Algoritmos <i>Swarm</i>	22
3.3. Descripción práctica	24
3.3.1. GUI_drone	25
3.3.2. GUI_swarm	27
4. Conclusiones	31

1. Resumen

La *Robótica de Enjambre* es un enfoque de la *Robótica Colectiva* que se inspira en los comportamientos autoorganizados de los animales sociales. El diseño de enjambres de robots se guía por los principios de *Inteligencia de Enjambre*. A través de reglas simples e interacciones locales, la robótica de enjambre tiene como objetivo diseñar comportamientos colectivos robustos, escalables y flexibles para la coordinación de un gran número de robots. Es decir, es el estudio de cómo diseñar grupos de robots que operan sin depender de ninguna infraestructura externa ni de ninguna forma de control centralizado. En un enjambre de robots, el comportamiento colectivo de los robots resulta de las interacciones locales entre los robots y el entorno en el que actúan.

A continuación se estudiará este campo de la robótica en detalle, de manera que en primer lugar se abordará una aproximación general de la Robótica de Enjambre, mostrando tanto un análisis teórico como un breve Estado del Arte; para luego centrarse en un caso de estudio particular.

2. Descripción de la tipología en el ámbito de la robótica móvil

En primer lugar se estudiará este campo desde un punto de vista teórico, realizando una introducción para contextualizarlo, así como analizando sus posibles aplicaciones y los ejemplos existentes actualmente.

2.1. Introducción

La Robótica de Enjambre es un campo de investigación que estudia cómo se pueden utilizar sistemas compuestos por múltiples agentes autónomos (robots) para realizar tareas colectivas, donde las tareas no pueden ser realizadas por cada robot individual por sí solo, o son realizadas más eficazmente por los robots como un grupo. Es decir, la Robótica de Enjambre es el estudio de cómo se puede diseñar un gran número de agentes relativamente simples encarnados físicamente de tal manera que un comportamiento colectivo deseado emerja de las interacciones locales entre los agentes y el medio ambiente.

De esta forma, los principios de la Inteligencia de Enjambre han sido ampliamente estudiados y aplicados a una serie de tareas diferentes en las que se utiliza un grupo de robots autónomos para resolver un problema con un enfoque distribuido, es decir, sin coordinación central. Pero antes, cabe estudiar el significado del término *Inteligencia de Enjambre*

2.1.1. Inteligencia de Enjambre

La mayoría de las investigaciones de Inteligencia de Enjambre se inspiran en cómo los enjambres de la naturaleza, como insectos sociales, peces o mamíferos, interactúan entre sí en el enjambre en la vida real. Estos enjambres varían en tamaño desde unos pocos individuos que viven en áreas naturales pequeñas hasta colonias altamente organizadas que pueden ocupar grandes territorios y constan de más de millones de individuos. Los comportamientos grupales que surgen en los enjambres muestran una gran flexibilidad y solidez, como la planificación de caminos, la construcción de nidos, la asignación de tareas y muchos otros comportamientos colectivos complejos en enjambres de diversas naturalezas. Los individuos en el enjambre de la naturaleza muestran habilidades muy pobres, sin embargo, los comportamientos grupales complejos pueden surgir en todo el enjambre, como la migración de multitudes de aves y bancos de peces, y la búsqueda de colonias de hormigas

y abejas. Es difícil para un individuo completar la tarea en sí, incluso un ser humano sin ciertas experiencias lo encuentra difícil, pero un enjambre de animales puede manejarlo fácilmente. Los investigadores han observado los comportamientos grupales inteligentes que emergen de un grupo de individuos con pocas habilidades a través de la comunicación local y la transmisión de información.

A continuación se muestran algunos casos de enjambres que se pueden encontrar en la naturaleza:

- **Colonias de bacterias:** Las bacterias a menudo funcionan como agregados multicelulares conocidos como biopelículas, intercambiando las señales moleculares para la comunicación entre células. Los beneficios comunitarios de la cooperación multicelular incluyen una división celular del trabajo, la defensa colectiva contra los antagonistas, el acceso a más recursos y la optimización de la supervivencia de la población al diferenciar los distintos tipos de células. La resistencia a los agentes antibacterianos de las bacterias en las biopelículas es 500 veces mayor que la de las bacterias individuales del mismo tipo.
- **Bancos de peces:** Los bancos de peces nadan en falanges disciplinadas y son capaces de fluir hacia arriba y hacia abajo a velocidades impresionantes y hacer un cambio sorprendente en la forma del banco sin colisionar como si sus movimientos fueran coreografiados. Los peces prestan mucha atención a sus vecinos cuando están en bancos con la ayuda de ojos a los lados de la cabeza y "marcas de bancos." en sus hombros. Los peces pueden beneficiarse de los bancos para buscar alimento y evitar a los depredadores.
- **Colonias de hormigas y abejas:** Las hormigas se comunican entre sí mediante feromonas, sonidos y tacto. Una hormiga con un intento exitoso deja un rastro marcando la ruta más corta en su regreso. Los senderos exitosos son seguidos por más hormigas, reforzando las mejores rutas e identificando gradualmente el mejor camino. Experimentos sugieren que las hormigas con mayor tasa de éxito intensifican sus intentos de búsqueda de alimento, mientras que las otras se aventuran en menos ocasiones o incluso cambian a otros roles.

Por último, el término *Inteligencia de Enjambre* fue originalmente concebido para denotar una clase de sistemas robóticos celulares. Más tarde, el término pasó a cubrir una amplia gama de estudios, desde la optimización hasta los estudios de insectos sociales, perdiendo su contexto de robótica mientras tanto. Recientemente, el término *Robótica de Enjambre* ha comenzado a utilizarse como la aplicación de inteligencia de enjambre a sistemas físicamente incorporados, y es el que se estudiará a continuación.

2.1.2. Robótica de Enjambre

Un enjambre de robots es un sistema multi-robot autoorganizado caracterizado por una alta redundancia. Las capacidades de detección y comunicación de los robots son locales y los robots no tienen acceso a la información global. El comportamiento colectivo del enjambre de robots surge de las interacciones de cada robot individual con sus compañeros y con el entorno. Por lo general, un enjambre de robots está compuesto por robots homogéneos, aunque existen algunos ejemplos de enjambres de robots heterogéneos.

Como aclaración, un grupo de robots homogéneos es aquel donde todos los robots son idénticos entre ellos y tiene las mismas funciones y capacidades, mientras que un grupo de robots heterogéneos es aquel que incluyen individuos con diferentes capacidades o características.

De esta forma, el principal problema de un sistema robótico de enjambre se puede enunciar de la siguiente manera: ¿Cómo se deben diseñar los individuos, tanto en términos de su encarnación física como de su control de comportamiento, de modo que un comportamiento deseado a nivel de enjambre surja de las interacciones entre los individuos como así como entre los individuos y el medio

ambiente? Este objetivo, que también puede considerarse como la “ingeniería de la autoorganización” en sistemas multi-robot, es una tarea difícil, si no imposible, de resolver en términos generales.

Se pueden definir un conjunto de criterios para distinguir la investigación en robótica de enjambres de disciplinas relacionadas:

- Los robots son autónomos, es decir, capaces de moverse e interactuar con el entorno sin un control centralizado; la tarea en cuestión puede ser realizada colectivamente por una gran cantidad de robots, lo que significa que el sistema debe diseñarse teniendo en cuenta la escalabilidad.
- El enjambre está formado por relativamente pocos grupos homogéneos de robots, y se centra en un gran número de individuos idénticos en lugar de equipos heterogéneos planificados centralmente donde cada individuo tiene un papel predefinido.
- Las capacidades de un solo robot (como las capacidades de detección, comunicación y computación) son limitadas en comparación con la dificultad de la tarea colectiva.
- Cada robot realiza la detección y la comunicación a nivel local, lo que garantiza que las interacciones entre los miembros del enjambre se distribuyan y no dependan de mecanismos de coordinación que dificultarían la escalabilidad.

La localidad de las interacciones y la comunicación, que podría verse como una limitación, tiene un efecto beneficioso sobre la escalabilidad y la solidez del sistema y, por lo tanto, generalmente se prefiere al uso de la comunicación y la detección globales. De esta forma, recordando la expresión *Inteligencia de Enjambre*, que ahora se usa ampliamente en el campo de la robótica de enjambre, se refiere a las capacidades superiores de un enjambre de agentes en comparación con sus individuos individuales. Los eventos locales desencadenados por los miembros del enjambre durante la ejecución de una tarea se traducen en un comportamiento global que a menudo trasciende las capacidades individuales, hasta el punto de que muchas tareas colectivas pueden ser realizadas con éxito por robots que no están programados explícitamente para ejecutar esas tareas: el global. Se dice que la dinámica macroscópica surge de las interacciones de los miembros del enjambre entre sí y con el entorno.

2.1.3. Propiedades

En la operación a nivel de sistema de un Enjambre Robótico debe exhibir tres propiedades funcionales que se observan en enjambres naturales y permanecen como propiedades deseables de los sistemas de múltiples robots.

- **Robustez:** El sistema robótico de enjambre debería poder operar a pesar de las perturbaciones del entorno o el mal funcionamiento de sus individuos. Se pueden observar una serie de factores en los insectos sociales detrás de la robustez de su funcionamiento. Primero, los enjambres son sistemas inherentemente redundantes; la pérdida de un individuo puede ser inmediatamente compensada por otro. En segundo lugar, la coordinación está descentralizada y, por lo tanto, es poco probable que la destrucción de una parte particular del enjambre detenga su operación. En tercer lugar, los individuos que componen el enjambre son relativamente simples, lo que los hace menos propensos al fracaso. Cuarto, la detección se distribuye; por tanto, el sistema es robusto frente a las perturbaciones locales del entorno.
- **Flexibilidad:** Los individuos de un enjambre deben ser capaces de coordinar sus comportamientos para abordar tareas de diferente naturaleza, pudiendo manejar un amplio espectro de entornos y condiciones operativas. La flexibilidad está habilitada por la naturaleza

distribuida y autoorganizada del enjambre. En un enjambre, los robots se asignan dinámicamente a diferentes tareas para cumplir con los requisitos del entorno específico y las condiciones operativas; además, éstos funcionan sobre la base de la detección y la comunicación locales y no dependen de una infraestructura preexistente ni de ninguna forma de información global. Por ejemplo, los individuos de una colonia de hormigas pueden encontrar colectivamente el camino más corto hacia una fuente de alimento o transportar una presa grande mediante la utilización de diferentes estrategias de coordinación.

- **Escalabilidad:** El enjambre debe poder operar en una amplia gama de tamaños de grupo y apoyar a una gran cantidad de personas sin afectar considerablemente el rendimiento. Es decir, los mecanismos de coordinación y las estrategias que se desarrollarán para los sistemas robóticos de enjambre deben garantizar la operación del enjambre bajo diferentes tamaños de enjambre. Idealmente, la introducción o eliminación de individuos no provoca un cambio drástico en el rendimiento del enjambre. La escalabilidad está habilitada por la detección y comunicación locales: siempre que la introducción y eliminación de robots no modifique drásticamente la densidad del enjambre, cada robot individual seguirá interactuando con aproximadamente la misma cantidad de pares, aquellos que están en su rango de detección y comunicación.

Se considera que las características antes mencionadas de la robótica de enjambre promueven la realización de sistemas tolerantes a fallas, escalables y flexibles.

2.1.4. Características

Ahora se resumirán las principales características distintivas de la investigación en Robótica de Enjambre.

- Primero, la investigación debe ser relevante para la coordinación de un enjambre de robots. Es decir, los individuos deben tener una encarnación física, estar situados y ser capaces de interactuar físicamente con su entorno. Además, los mecanismos de coordinación que se están estudiado debería prometer ser escalable para una amplia gama de tamaños de enjambre.
- En segundo lugar, el sistema robótico que se estudia debe ser bastante homogéneo. Es decir, los individuos que componen el enjambre deberían ser bastante idénticos, al menos a nivel de interacciones. Las estrategias de coordinación desarrolladas para sistemas heterogéneos de múltiples robots, que consisten en individuos que difieren en sus interacciones debido a su forma física o su control de comportamiento, quedan fuera del enfoque de la robótica de enjambre.
- Tercero, los individuos deben ser relativamente simples. El criterio de simplicidad en la definición no se refiere directamente a la complejidad del hardware y software de los robots, sino que pretende enfatizar las limitaciones en sus capacidades individuales en relación con la tarea. Los miembros del sistema de enjambre deben ser relativamente incapaces o ineficientes por sí mismos con respecto a la tarea en cuestión. Es decir, la tarea debe ser difícil o imposible de realizar por un solo robot, y la cooperación de un grupo de robots debe ser esencial, o el despliegue de un grupo de robots debe mejorar el rendimiento y robustez del manejo de la tarea.
- Cuarto, los individuos deben tener habilidades de interacción local. Esta restricción garantiza que la coordinación entre los robots se distribuya y que sea más probable que se amplíe con el tamaño del enjambre. Es probable que los mecanismos que se basan en capacidades de interacción global estén limitados por el ancho de banda y el rango del canal de comunicación y pueden crear mecanismos de coordinación no escalables.

2.2. Clasificación

En cuanto a la clasificación dentro del campo de la Robótica de Enjambre, podemos tomar dos puntos de vista distintos para diferenciar los sistemas existentes: los métodos de modelado empleados, o el tipo de comportamiento colectivo (suponiendo que se emplea un *modelo a partir de algoritmos de robótica de enjambre*).

2.2.1. Métodos de modelado

El modelado es un método utilizado en muchos campos de investigación para comprender mejor los aspectos internos del sistema que se investiga. El modelado ayuda a la Robótica de Enjambre ya que se supone que un algoritmo robótico de enjambre es escalable a cientos de miles de robots en la población. El tiempo y el dinero son limitados para tal escala de experimentos, por lo que interesa que los experimentos se puedan hacer de una manera más fácil.

Teniendo en cuenta las características de la robótica de enjambre, los métodos de modelado se dividen en cuatro tipos: basada en sensores, microscópica, macroscópica y basada en inteligencia de enjambre. Los cuatro métodos se describen en detalle en esta sección.

- **Modelado basado en sensores:** En el método de modelado basado en sensores, los sensores y actuadores de los robots se modelan como los componentes principales del sistema junto con los objetos del entorno. Luego, las interacciones de los robots se modelan de la manera más realista y simple posible. Este método de modelado se usa de manera principal, y es el método más antiguo para experimentos robóticos. Sin embargo, las investigaciones anteriores que utilizaron métodos de modelado basados en sensores no tuvieron en cuenta las limitaciones físicas reales, ahora los investigadores introducen el principio físico en el modelo.
- **Modelado microscópico:** En el modelado microscópico, los robots y las interacciones se modelan como una máquina de estados finitos. Los comportamientos de cada robot se definen como varios estados, y las condiciones de transferencia se basan en la entrada de comunicación y detección. Dado que el modelo se basa en los comportamientos de cada robot, la simulación debe ejecutarse varias veces para obtener los comportamientos promedio del enjambre. En la mayoría de las investigaciones de robótica de enjambre, se utiliza el modelo microscópico probabilístico, ya que el ruido se puede modelar como probabilidad en el modelo. En un modelo microscópico probabilístico, las probabilidades se valoran a partir de los experimentos de robots reales, y el modelo se itera con estas probabilidades de transferencia de estado en la simulación para predecir el comportamiento del enjambre.
- **Modelado macroscópico:** El modelado macroscópico es un método de modelado opuesto al modelado microscópico. En el modelado macroscópico, el comportamiento del sistema se define como una ecuación diferencial y un estado del sistema representa el número promedio de robots en este estado en el paso de tiempo. La principal diferencia entre los modelos microscópicos y macroscópicos es la granularidad de los modelos. El modelo microscópico para el comportamiento a nivel individual se utiliza para simular los comportamientos grupales, mientras que el modelo macroscópico simula los comportamientos a nivel de enjambre. El modelo microscópico itera el comportamiento del enjambre y el modelo macroscópico puede dar el estado final del enjambre. De esta forma, el modelo macroscópico puede tener una visión global del enjambre mientras que el modelo microscópico puede mostrar los detalles de los comportamientos del enjambre. Los modelos macroscópicos probabilísticos también son ampliamente utilizados por los investigadores.

- **Modelado a partir de algoritmos de inteligencia de enjambre:** Existen varios comportamientos diferentes de enjambre. Este enfoque para el modelado ha ido ganando popularidad con el tiempo, hasta ser el principal a día de hoy. Por ello, si se sobreentiende que se emplea este tipo de modelado, se puede realizar otra clasificación en función del tipo de comportamiento colectivo. Esta otra clasificación se estudiará en detalle a continuación.

2.2.2. Tipos de comportamiento colectivo

Un sistema de Enjambre Robótico puede presentar diferentes tipos de comportamientos colectivos, por lo tanto, se podrían clasificar los sistemas en función del conjunto de comportamientos que puedan desarrollar. A continuación, se detallan varios de estos comportamientos colectivos. Algunos de los comportamientos, como la agregación y el movimiento colectivo, son bastante básicos y constituyen un nivel previo para tareas más complejas. Se presentan en orden creciente de complejidad.

- **Agregación:** La agregación autoorganizada, es decir, la tarea de reunir un número de individuos autónomos en un lugar común, es un comportamiento básico ampliamente observado en la naturaleza con muchas especies animales. Se han propuesto varios modelos matemáticos para describir la agregación y se han diseñado sistemas robóticos con varios algoritmos para implementar dinámicas de agregación. Esta tarea se ha estudiado como un problema independiente o en el contexto de tareas más especializadas que requieren reunir a múltiples agentes.
- **Flocado:** El agrupamiento es un comportamiento observado en la naturaleza en muchas especies de aves, que forman grandes grupos de individuos que se mueven juntos hacia una ubicación objetivo común. Otros ejemplos de comportamientos colectivos análogos que se encuentran en los animales son los bancos de peces. Estos comportamientos emergen a nivel colectivo de manera distribuida, como consecuencia de interacciones locales entre agentes autónomos, y como tales son de interés para los investigadores de robótica de enjambres, quienes han estudiado los mecanismos en la base del comportamiento animal y han tratado de replicar el agrupamiento en enjambres robóticos. En la mayoría de los trabajos existentes, los robots con capacidades de detección limitadas deben mantener una formación compacta midiendo la distancia y la orientación relativa de sus vecinos; los casos en los que uno o grupos de robots están fuera del rango de detección y comunicación del resto de los robots generalmente no se consideran en los estudios de flocado, donde la suposición habitual es que todos los robots tienen al menos un vecino que los conecta con el resto del enjambre.
- **Búsqueda de alimento:** La tarea de forrajeo colectivo, inspirada en el comportamiento de las hormigas en colonias, es otro escenario comúnmente estudiado en la robótica de enjambres. Las hormigas y otros animales sociales pueden explotar de manera eficiente las fuentes de alimentos mediante interacciones locales entre individuos. En un sistema robótico de enjambre artificial para la tarea de búsqueda de alimento, se designa un área específica como el "nido", y el objetivo del enjambre es encontrar elementos dispersos en el entorno y llevarlos al nido. El forrajeo múltiple es una extensión de la tarea de forrajeo en la que se deben recolectar diferentes tipos de artículos y cada artículo se entrega a un nido específico para el tipo de artículo. Las aplicaciones prácticas de este tipo de tareas incluyen desminado, limpieza de desechos peligrosos, búsqueda y rescate y exploración planetaria.
- **Agrupación y clasificación de objetos:** La agrupación de objetos se refiere a una tarea en la que los objetos dispersos en el entorno deben agruparse. En comparación con la búsqueda de alimento, en la tarea de agrupamiento de objetos no hay un lugar de destino predefinido para los objetos recolectados, el objetivo es colocar los objetos cerca uno del otro. En una variación

de esta tarea, hay más de un tipo de objetos y los grupos deben formarse por separado para cada tipo de objeto; en este caso, la tarea suele denominarse clasificación, porque los objetos se clasifican según su tipo.

- **Navegación:** Un escenario de navegación colectiva es aquel en el que un robot con capacidades limitadas de detección y localización puede alcanzar un objetivo en una ubicación desconocida con la ayuda de otros robots. Los estudios donde múltiples robots deben navegar a la misma ubicación no se consideran en esta categoría, donde el foco está en escenarios donde la ubicación de destino debe ser alcanzada por un solo robot, que puede aprovechar la presencia de los otros robots para facilitar su tarea.
- **Formación de caminos:** La formación de rutas en la robótica de enjambres se refiere a un proceso en el que los robots pueden construir colectivamente una ruta entre dos ubicaciones en el entorno, de modo que se minimice el tiempo necesario para llegar a una ubicación desde la otra. Esta tarea también puede denominarse formación de cadena, porque a menudo el camino está marcado por una cadena de robots, ya sea estacionarios o en movimiento. Como se describe en la sección dedicada a la tarea de búsqueda de alimento, los mecanismos de formación de caminos se pueden observar en muchos casos en el contexto de búsqueda de alimento, porque a menudo los robots comparten ubicaciones de interés y, por lo tanto, pueden beneficiarse al compartir información sobre cómo llegar a esas ubicaciones.
- **Despliegue:** En un escenario de autoimplementación, los robots deben desplegarse en un entorno sin coordinación central. Esta tarea tiene potencialmente muchas aplicaciones prácticas, que van desde el mapeo de entornos desconocidos hasta sistemas de vigilancia autónomos.
- **Manipulación colaborativa:** Por lo general, los sistemas de enjambre permiten a los agentes ejecutar tareas colectivas de manera más eficiente que lo que puede hacer cada individuo por sí solo; en algunos casos, la tarea en cuestión no puede ser ejecutada por un solo individuo, sino que requiere la cooperación entre múltiples individuos. Un ejemplo típico tomado de las sociedades de insectos es la recuperación de grandes artículos de comida por parte de grupos de hormigas: dependiendo del tamaño del artículo, esta tarea puede requerir una gran cantidad de hormigas, que deben trabajar en coordinación para llevar a cabo la tarea con éxito (es decir, transportar el alimento al nido). La observación del comportamiento de las hormigas ha demostrado que dicha coordinación se logra sin un control central y con cada hormiga usando reglas simples gobernadas por interacciones locales con otras hormigas y con el medio ambiente. Aunque es posible que la tarea no se complete con una eficiencia óptima (por ejemplo, pequeños grupos de hormigas pueden realizar múltiples intentos fallidos antes de reclutar otras hormigas).
- **Asignación de tareas:** La asignación de tareas o la división del trabajo en un sistema de robótica de enjambre se refiere a la capacidad de cambiar dinámicamente la tarea ejecutada por cada robot en función de la percepción local del entorno. Con esta capacidad, los sistemas robóticos pueden exhibir dinámicas de trabajo eficientes al adaptar la proporción de robots que participan en una tarea determinada (o que no participan en ninguna tarea) en función de la demanda actual de la tarea o la ganancia esperada de la ejecución de la tarea. Aunque los robots están dotados de capacidades de detección local y, por lo tanto, no pueden medir directamente el estado global del medio ambiente, las interacciones locales entre los robots y el medio ambiente se pueden utilizar para adaptar el comportamiento de los robots individuales con el fin de beneficiar la eficiencia del enjambre.

■ **Otras tareas de robótica de enjambre:**

- Localización de fuentes de olor
- Ensamblaje de objetos
- Autoensamblaje y morfogénesis
- Movimiento coordinado
- Estimación del tamaño del grupo
- Encuentro distribuido
- Toma de decisiones colectiva
- Interacción humano-enjambre

2.3. Aplicaciones

Ejemplos de tareas que podrían abordarse de manera rentable utilizando la robótica de enjambres son el desminado, la búsqueda y el rescate, la exploración planetaria o submarina y la vigilancia.

En cuanto a las aplicaciones, la robótica de enjambre ha acumulado ya una colección de problemas estándar que se repiten a menudo en la literatura. Un grupo de problemas se basa en la formación de patrones: Agregación, auto-organización en un entramado, despliegue de antenas distribuidas o matrices distribuidas de sensores, cobertura de zonas, cartografía del entorno, despliegue de mapas, creación de gradientes, etc. La agregación sirve para realizar otras tareas, como movimiento colectivo, el autoensamblaje y la formación de patrones, o intercambiar información, los robots deben reunirse inicialmente. Este problema de agregación ha sido estudiado desde un enfoque de enjambres robóticos. Asimismo junto a la agregación se destaca el problema de la dispersión, el cual consiste en distribuir los robots a lo largo del espacio con la idea de cubrir el mayor área posible sin perder conexión entre los robots que componen el enjambre. Este problema se emplea para generar que el enjambre actúe como un sensor colectivo o para tareas de exploración. Los robots enjambre pueden realizar tareas en las que el objetivo principal es cubrir una amplia región. Mediante la dispersión, como ya se ha comentado, los robots tienen la capacidad de realizar tareas de vigilancia, por ejemplo, en bosques, lagos, etc. Puede ser realmente útil para detectar eventos peligrosos, como una fuga de una sustancia química. La principal ventaja sobre una red de sensores es que el enjambre puede moverse y centrarse en el problema e incluso actuar para prevenir las consecuencias de ese problema. De este modo, los enjambres de robots pueden ser realmente útiles para tareas peligrosas. Por ejemplo, para la detección de minas y limpieza. Puede ser más útil que un único robot especializado, sobre todo por la robustez del enjambre: Si un robot falla y la mina explota, el resto del enjambre sigue trabajando, con un solo robot esto no es posible.

Un segundo grupo de problemas se centra en alguna entidad específica del entorno: búsqueda de objetivos, localización de la fuente de una pluma química, búsqueda de alimento, recuperación de presas, etc. Y otro grupo de problemas se ocupa del comportamiento grupal más complejo: transporte cooperativo, minería (recogida de palos), pastoreo, rebaño, contención de derrames de petróleo, etc. Por supuesto, esta lista no es exhaustiva; hay otras tareas robóticas genéricas de la robótica, como la evitación de obstáculos y la navegación en todo el terreno, también se aplican a los enjambres.

Otro aspecto de la robótica de enjambre que hay que mencionar es el control del enjambre. En última instancia, una vez ideados los algoritmos para la realización de las tareas la realización práctica requiere robustez y ésta es el resultado de un control adecuado. El control de los enjambres plantea nuevos retos a los ingenieros de robótica. El ejemplo más clásico, desde el punto de vista de la ingeniería de control, es quizás el control de la formación, cuya base reside en crear una

figura global cambiando las posiciones de los robots. Esto se aplica por ejemplo en el control de equipos multirobots o de aviones o vehículos acuáticos autónomos. Estos estudios llevan a considerar problemas de estabilidad asíncrona de sistemas distribuidos y están muy en línea con el impulso original hacia la robótica de enjambre.

Actualmente no existen aplicaciones comerciales reales aplicaciones comerciales. Las razones para ello son variadas, dentro de las cuales existen las siguientes:

- Diseño de algoritmos. La robótica de enjambre debe diseñar tanto los robots físicos como los comportamientos de los robots individuales, de modo que el comportamiento colectivo global surja de sus interacciones. Por el momento, no existe ningún método general para pasar de los comportamiento individual al comportamiento colectivo.
- Implementación y prueba. El uso de muchos robots reales necesita de una buena infraestructura de laboratorio para poder realizar experimentos.
- Análisis y modelado. Los sistemas robóticos de enjambre suelen ser estocásticos y no lineales, por lo que es difícil crear modelos matemáticos para la validación y la optimización es difícil. Estos modelos pueden ser necesarios para crear aplicaciones de seguridad en el mundo real.

2.4. Ejemplos de robots con aplicación en la robótica de enjambre

2.4.1. I-Swarm

La visión fundamental del proyecto propuesto es la de un enjambre de microrobots, capaz de realizar tareas que no son posibles con un solo microrobot o con un pequeño grupo de microrobots. El resultado esperado es la realización de un efecto de autoorganización observable en el enjambre de robots, similar al observado en otros sistemas ecológicos como los estados de hormigas, las colonias de abejas y otras agregaciones de insectos.

Los beneficios potenciales de un sistema de este tipo son muchos, como la mayor flexibilidad y adaptabilidad del sistema al entorno, la robustez ante los fallos, etc. Además, su comportamiento colectivo abre nuevos campos de aplicación, que no pueden resolverse con las herramientas actuales. Con un sofisticado sistema de posicionamiento adecuado, posiblemente basado en el utilizado por los insectos y que incorpore sensores táctiles y un sistema de visión pequeño pero eficaz, los agentes individuales podrán comunicarse entre sí y, de este modo, permitir y promover el deseado efecto de enjambre.

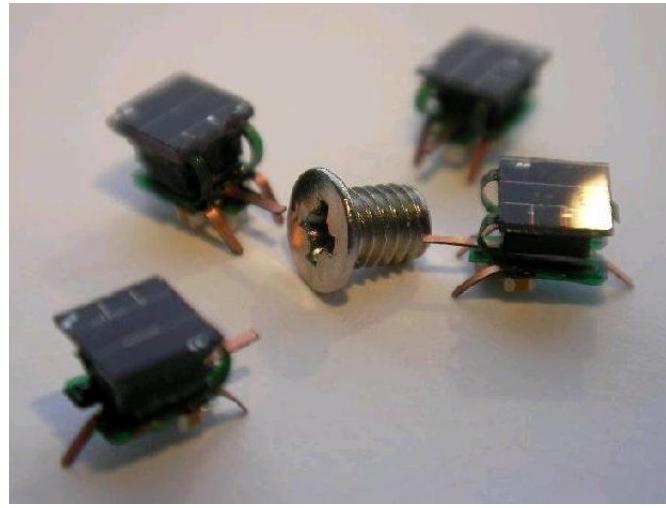


Figura 1: Robot I-Swarm

2.4.2. iRobot Swarm

El *iRobot Swarm* se compone de 112 robots que trabajan en conjunto para conseguir objetivos grupales. Estos robots tienen la capacidad de cargarse dirigiéndose a la estación de carga correspondiente de forma autónoma, que junto a la baliza que les brinda una navegación de larga distancia, permiten realizar múltiples tareas grupales. Dentro de sus características destaca su pequeño tamaño, el conjunto de sensores que los engloba junto al *hardware* de comunicaciones y los dispositivos con interfaces de interacción humano-robot.

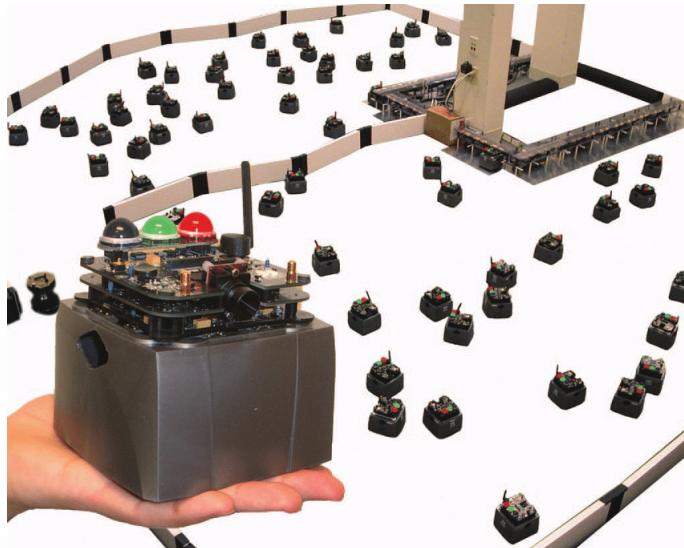


Figura 2: iRobot Swarm Robots

2.4.3. Alice Mobile Robot

Alice tiene un cuerpo de prisma rectangular, con dimensiones de 21x21x21mm y hecho de plástico y PCB. Para la locomoción se utilizan dos motores Swatch de alta eficiencia. Alice tiene muchos módulos sensoriales opcionales de proximidad IR para la detección de obstáculos y el módulo de comunicación de robot a robot de corto alcance, módulo de recepción IR, módulo de cámara lineal, módulo de comunicación por radio (RF) y módulo de extensión ANT. Además, existe un módulo de agarre opcional y varios módulos de locomoción para diferentes terrenos. Se informa de diez horas de autonomía con dos pilas de botón y veinte horas de autonomía se consiguen con una batería LiPo adicional.

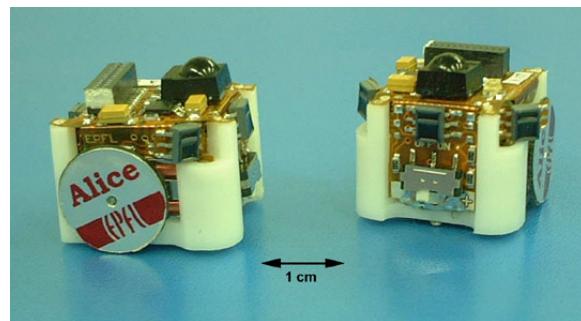


Figura 3: Alice Mobile Robot

2.4.4. Jasmine Mobile Robot

Jasmine es otra plataforma micro-robot que tiene un cuerpo de prisma rectangular de 26x26x26 mm y está hecha de aluminio y PCB. Para la locomoción se utilizan dos pequeños motores con cabeza de engranaje. Jasmine tiene 6 sensores IR(LED+receptor) para la proximidad y la comunicación local con los robots. Tiene un potente LED IR para el análisis detallado de un objeto de interés y un módulo de comunicación IR con el host. Jasmine III tiene un diseño modular en el que diferentes módulos de detección como el sensor de luz ambiente, sensor de color y diferentes módulos de locomoción. Dos horas de autonomía con baterías LiPo.

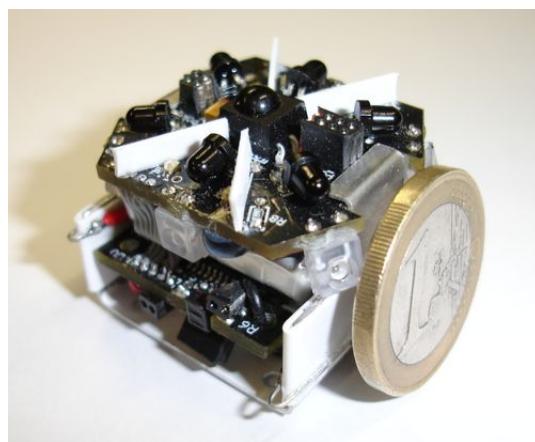


Figura 4: Jasmine Mobile Robot

2.4.5. Kilobot project

El proyecto Kilobot pretende diseñar un sistema robótico para probar los algoritmos colectivos con una población de cientos o miles de robots. Cada robot está fabricado con piezas de bajo coste y tarda 5 minutos en estar completamente montado. El sistema también proporciona varias operaciones generales para un gran enjambre, como actualización de programas, encendido, carga de todos los robots y volver a casa.

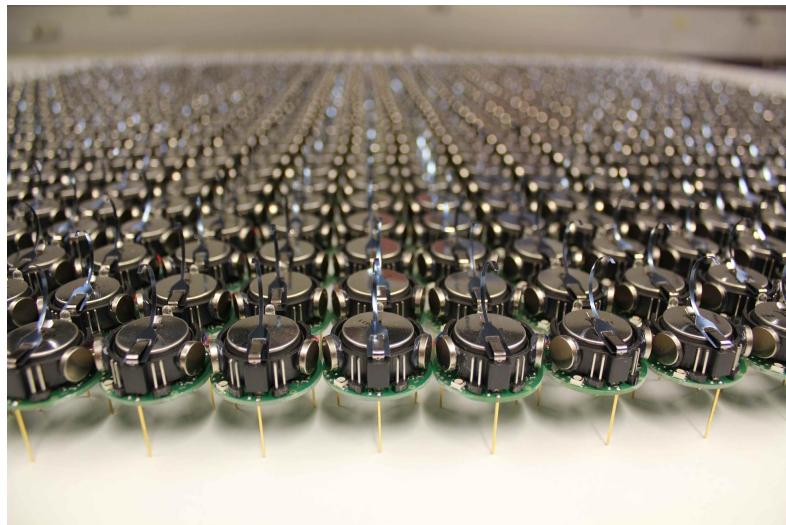


Figura 5: Kilobot project

2.4.6. E-Puck Education Robot

E-puck es un robot móvil educativo de sobremesa desarrollado en la *Ecole Polytechnique Fédérale* de Lausana (EPFL) para una amplia explotación en actividades de enseñanza. Los principales objetivos de este desarrollo fueron:

- El uso de una plataforma común en todos los cursos de la EPFL relacionados con la robótica móvil, para sustituir a los diferentes robots que se utilizaban anteriormente.
- El uso de un robot móvil en cursos no relacionados con la robótica, por ejemplo, procesamiento de señales, control automático y programación embebida, con el fin de proponer más ejercicios basados en proyectos.
- La introducción de robots móviles en una fase más temprana del en el plan de estudios, lo que implica el despliegue en clases más grandes (de 50 a 100 alumnos).

Estos robots tienen una estructura mecánica muy sencilla de entender, operar y mantener. Son baratos y flexibles, son capaces de cubrir un gran abanico de actividades educativas gracias a sus potentes sensores, poder de procesamiento y sus extensiones.



Figura 6: E-Puck Educationa Robot

3. Caso de estudio: Robótica de Enjambre aérea

Una vez realizado el análisis teórico, se llevará a cabo un estudio más práctico sobre una parte de la robótica de enjambre: la *robótica de enjambre aérea*. Se hará una introducción acerca de este ámbito para luego centrarse en un ejemplo concreto del mismo.

3.1. Introducción

3.1.1. Vehículos aéreos no tripulados (UAV)

Los *vehículos aéreos no tripulados* son aeronaves que se caracterizan principalmente por no llevar a bordo piloto que los controle. El control se puede realizar de forma remota o directamente se mueve de forma autónoma. En comparación con los vehículos aéreos tripulados, son más sencillos y ligeros, su legislación es mucho más laxa y su pérdida, en principio, no implica la pérdida de vidas humanas por lo que su uso puede estar enfocado a misiones que entrañan algún tipo de riesgo.



Figura 7: Ejemplo de UAV.

En los últimos tiempos la popularidad de estas máquinas se ha disparado, tanto a nivel profesional como a nivel aficionado. Es sencillo adquirir un cuadricóptero (aeronaves con cuatro alas rotatorias) con una gran estabilidad y autonomía a precios relativamente bajos (por debajo de 500

euros). Además, algunos fabricantes como Parrot o DJI han creado librerías que permiten desarrollar programas propios para controlarlos. Por este motivo, muchos centros educativos, empresas y particulares se han interesados en ellos para poder implementar tareas específicas.

Sumando la simplicidad mencionada anteriormente con el bajo coste, es normal que aparezca la idea de utilizar varios aparatos de forma conjunta para realizar ciertas tareas. Por ejemplo, si se quiere rastrear una zona catastrófica en busca de supervivientes las opciones de éxito aumentan al utilizar más de un aparato. Otro ejemplo podría ser la vigilancia de una gran zona, donde prácticamente es obligatorio el uso de más de un agente.

3.1.2. Enjambres aéreos

Un *enjambre de UAV* se puede definir como un conjunto de éstos que vuelan de acuerdo a un modelo inspirado biológicamente en el vuelo de las bandadas de pájaros y/o el comportamiento de los insectos sociales. En comparación con los enjambres formados por vehículos terrestres, el estudio de los enjambres en UAV llegó unos veinte años más tarde, en la década de los noventa. Los investigadores buscan que un conjuntos de pequeños y simples UAV tengan prestaciones similares a las que se consiguen con grandes UAV, siendo capaces de realizar tareas reservadas hasta ahora sólo a vehículos de gran tamaño.



Figura 8: Enjambre de UAV.

Las características de este tipo de enjambres son muy similares a las mencionadas anteriormente cuando se ha definido la Robótica de Enjambre:

- Agentes autónomos que forman el enjambre.
- Capacidad para conectarse con otros agentes del enjambre, generalmente con los más cercanos.
- Comportamiento individual siguiendo unas directrices básicas.
- Inexistencia de un control centralizado del enjambre.

Para poder llevar a cabo un enjambre de UAV es necesario que éstos sean totalmente autónomos en vuelo, sin necesidad de control remoto. En esta evolución el elemento crítico es el piloto automático, ya que a nivel comercial no existen pilotos automáticos que permitan la navegación autónoma del aparato y menos para UAV de pequeño tamaño. En la actualidad, los sistemas empleados llevan a cabo una navegación hacia una serie de puntos que se han programado con anterioridad. Los elementos que se deberían desarrollar para conseguir un vehículo autónomo en vuelo son los siguiente:

- Piloto automático avanzado capaz de reaccionar de forma rápida a las entradas que reciban los elementos de control del vehículo.
- Controlador de vuelo con prioridad en tomar el control del vehículo y que se encargue de tareas como evitar colisiones.
- Mecanismo de arbitraje que regule el control del piloto automático y el controlador de vuelo mencionados anteriormente.

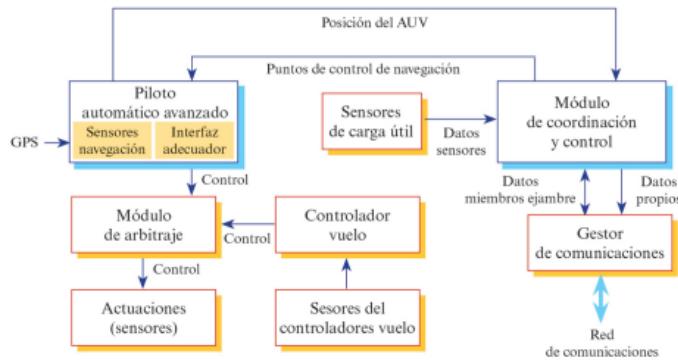


Figura 9: Sistema de control de un vehículo autónomo avanzado.

En lo que respecta a las comunicaciones, la red encargada de ello debe ser lo suficientemente robusta para evitar que haya retardos en la realización de ciertas tareas, además de tener la capacidad de recuperación y reconfiguración en caso de pérdida de vehículos que estaban conectados a la red. Están en desarrollos algoritmos de redes de comunicaciones basadas en *clusters* de nodos con capacidad de autoconfiguración y robustos frente a degradaciones en cuanto a la operatividad de los nodos que la forman. Este tipo de redes están basados en lo que se conoce como MANET (Mobile and Ad Hoc Wireless Networks). El conjunto de nodos está formado por un nodo central, otros nodos que forman el conjuntos y otros que sirven para conectarse con otros conjuntos. Esta arquitectura de comunicaciones es de tipo modular y formada por múltiples capas.

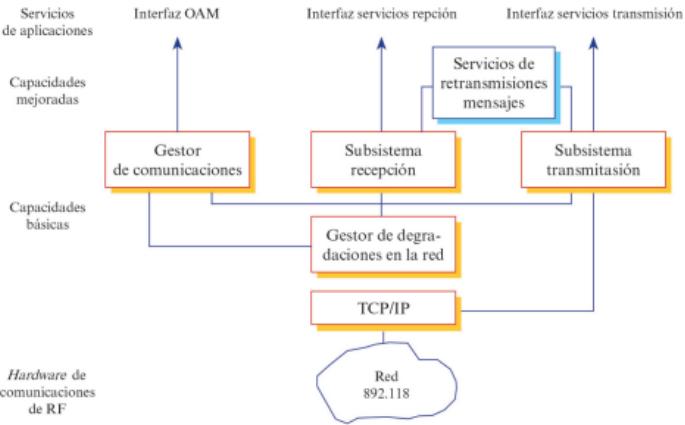


Figura 10: Arquitectura de comunicaciones de un enjambre de UAV.

Existen diversos modelos de gestión y control sobre el comportamiento de un enjambre de UAV. En los siguientes apartados se explicará cada uno de ellos:

- *Maestro y subordinados*: unos de los UAV, normalmente el de mayor tamaño y carga útil, utiliza un modelo centralizado de control, estando en contacto con una estación de control en tierra de la cual recibe órdenes y a la que manda resultados. A su vez, este UAV estará conectado a otros UAV de menor tamaño que también estarán conectados entre sí. Por lo general, los UAV pequeños serán los que lleven a cabo la tarea a partir de la información aportada por el UAV maestro, que estará equipado con los radares y sensores necesarios.

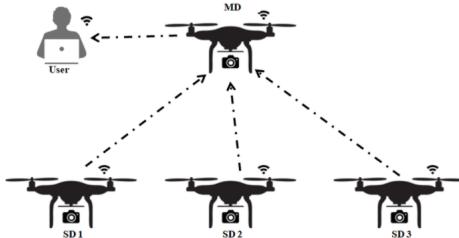


Figura 11: Estructura Master-slaves.

- *Trabajo en equipo*: busca la realización de tareas complejas utilizando la cooperación de pequeños UAV de bajo coste y capacidad limitada de procesos. Los UAV pertenecientes al enjambre no tienen por qué ser iguales ni tener las mismas cargas útiles. Estarán capacitados para tener un comportamiento autónomo tanto en navegación como en tareas (la forma en la que hacer la tarea estará preprogramada), y además, en caso de pérdida de algún componente del enjambre, otro podría reemplazarle en la realización de la tarea.
- *Modelo de agentes inteligentes*: este modelo está basado en UAV individuales que tienen cierta inteligencia, es decir, serán capaces de tomar decisiones relacionadas con la tarea asignada. Cada UAV individual se caracteriza como una máquina de estados finitos, e irá cambiando de estado en función de eventos que vayan sucediendo, bien en el propio UAV o en el entorno.

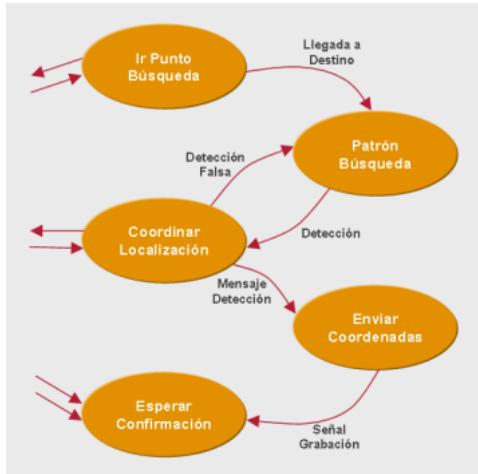


Figura 12: Algoritmo de inteligencia de cada agente.

- *Modelo de campos computacionales*: este modelo está basado en el comportamiento de algunos enjambres de insectos, con especial atención en las hormigas. Los campos computacionales se definen como estructuras de datos que representan una abstracción del mundo, que será lo que perciban los componentes del enjambre y por otros agentes llamados "de influencia" que definen el comportamiento de los miembros del enjambre. En este modelo el enjambre se considera como un todo, y cada componente realiza una tarea que sólo tiene sentido en el contexto del comportamiento colectivo. Se alcanzará el objetivo, no por las tareas individuales, sino por el comportamiento colectivo.



Figura 13: El comportamiento de las hormigas sirvió de inspiración.

Las aplicaciones de los enjambres de UAV son muy variadas: tienen una gran aplicación en el ámbito militar, también se usan en aplicaciones de seguridad y en el campo de la agricultura y el medio ambiente.

- *Aplicaciones ISTAR*: se incluyen las aplicaciones militares orientadas a la localización, identificación, seguimiento y asignación de blancos u otros objetos. Los agentes del enjambre se distribuirán inicialmente con el objetivo de detectar un blanco para posteriormente poder identificarlo y seguirlo si éste se moviese.



Figura 14: Uso militar de UAV.

- *Búsqueda y ataque*: aquellas aplicaciones militares en las que el enjambre se encargará de eliminar elementos activos del enemigo para destruirlos. En este caso, unos miembros del enjambre se encargarán de la detección del blanco activo y otro se encargarán del ataque al mismo o de mandar esa información a un centro de mando para que actúasen en consecuencia.



Figura 15: Enjambre de UAV en búsqueda de objetivos.

- *Protección civil*: serían aquellas aplicaciones en situaciones de peligro para el ser humano, como pueden ser desastres relacionados con la energía atómica, monitorización de zonas de desastres naturales para evaluar daños y buscar supervivientes, monitorización de eventos multitudinarios o el tráfico, etc.



Figura 16: Enjambre de UAV monitorizando la ciudad.

- *Agricultura*: existen numerosas aplicaciones en este ámbito para el uso de enjambres de UAV. Por ejemplo, el control de plagas en cultivos o la monitorización del correcto desarrollo de los mismos. También se pueden utilizar para el reparto de abono, la creación de mapas de superficies cultivadas o la detección de plantaciones ilegales.



Figura 17: Uso de enjambres UAV en plantaciones.

- *Vigilancia medioambiental*: en este campo, los enjambres de UAV podrían utilizarse para adentrarse en zonas peligrosas como un volcán en erupción para analizar los efectos. También se podría usar para análisis de aire, contaminación en mares o ríos, etc.



Figura 18: Incursión de un UAV en zona de erupción de un volcán.

- *Meteorología*: los UAV permiten el acceso a zonas donde los vehículos tripulados no pueden

adentrarse.

3.2. Descripción teórica

Entre todas las simulaciones de enjambres de drones disponibles, es posible determinar de que la mayoría de estas requiere un proceso de instalación bastante laborioso, en el cual es bastante determinante la versión de ROS instalada y/o las capacidades del dispositivo en el cual se realizan las simulaciones. Además, los usuarios que se inician en el campo de la investigación de la robótica de enjambre mediante simulaciones de drones a menudo necesitan interactuar con múltiples lenguajes de programación. Por lo que una alternativa a todo este farragoso proceso es el proyecto desarrollado por Enrica Soria, Fabrizio Schiano y Dario Floreano; SwarmLab, un software implementado única y exclusivamente en MATLAB, el cual tiene como objetivo la creación de procesos y métricas estandarizadas para cuantificar el rendimiento y la robustez de los algoritmos de los enjambres, y en particular, se centra en los drones.

El entorno de programación escogido fue MATLAB debido a que este entorno de desarrollo ofrece las siguientes posibilidades:

- **Es un lenguaje de alto nivel** el cual facilita el uso de esta herramienta a usuarios con escasos conocimientos de programación.
- **Hay una gran cantidad de toolboxes** disponibles para el diseño, control, análisis y visualización en la plataforma de *Mathworks* reduce el esfuerzo y la complejidad del entorno de *SwarmLab*.
- **El código implementado en MATLAB se traduce automáticamente a C/C++** reduciendo el coste computacional del mismo e incluso facilitando la interconexión de este junto a sistemas empotrados.

A parte del uso del entorno de programación de MATLAB, el *SwarmLab* fue desarrollado a partir del paradigma de Programación Orientada a Objetos (OOP), puesto que esta permite una mejor modularización del software, determinando distintas clases y métodos que fuesen útiles tanto en conjunto como por separado.

Las clases creadas son la clase *Drone* y la clase *Swarm*, ambas se encuentran determinadas de la siguiente manera.

3.2.1. Clase *Drone*.

La clase *Drone* representa el bloque de construcción para simular un enjambre. Esta clase admite la definición de *cuadricópteros* o *drones de ala fija*. Una instancia de *Drone* se define por:

- **los parámetros relacionados con la plataforma elegida** (por ejemplo, la masa parámetros aerodinámicos y de control),
- **Un vector de estados** el cual se encuentra representado por las coordenadas de la posición norte, este y abajo en el marco inercial, la velocidad lineal medida a lo largo de los ejes x, y, z del marco del cuerpo, tres ángulos de Euler que describen la orientación del dron, es decir, los ángulos *roll*, *pitch* y *yaw*; y las velocidades angulares medidas en el marco del cuerpo. En total un vector de 12 componentes que determinan el estado del dron.
- **Las variables del path**, las cuales incluye la lista de *waypoints* que deben de ser recorridos para completar correctamente la ruta previamente planificada.

- **Las variables gráficas** que permiten la visualización del dron junto a sus propiedades intrínsecas.

Además de estas variables o propiedades esta clase contiene métodos que permiten la creación de nuevas instancias de la clase, métodos relacionados con los cálculos de la cinemática y dinámica del dron en base a sus parámetros físicos, métodos de control del dron a través de dos pilotos autónomos; o incluso, métodos que permiten el control y navegación de un dron único; permitiendo el manejo manual del mismo.

3.2.2. Clase *Swarm*.

La clase *Swarm* contiene las propiedades y métodos necesarios para instanciar, inicializar y controlar los objetos de tipo *Swarm*. Considerando esta clase como un conjunto de instancias de la clase *drone*. Sus propiedades fundamentales son:

- **Drones:** un vector de objetos de tipo *Drone*.
- **Número de agentes** determina el número de objetos o *Drones* incluidos en el *Swarm*.
- **El algoritmo de navegación**, puesto que existen diversos tipos de algoritmos de navegación que pueden ser utilizados, en esta propiedad se selecciona el que va a ser utilizado en cada *Swarm*.

3.2.3. Algoritmos *Swarm*.

En SwarmLab, se encuentran implementados y adaptados dos algoritmos representativos pertenecientes a la categoría de enjambre descentralizado. La razón de esta elección es que un enfoque descentralizado puede hacer que el sistema sea fácilmente escalable y robusto ante los fallos de un solo individuo. El primer algoritmo es obra de **Olfati-Saber**, el cual propone un marco teórico formal para el diseño y el análisis de algoritmos de enjambre basado en los campos de potencial y la teoría de grafos. Se basa en la construcción de un potencial colectivo que penaliza la desviación de los agentes con respecto a una forma reticular. Además, de un término de consenso que hace que los agentes se pongan de acuerdo en la dirección de su velocidad. Mediante el uso de este algoritmo, en situaciones donde todos los drones se encuentran en equilibrio y existe una ausencia de obstáculos, estos ocupan posiciones a una distancia constante de sus vecinos y se trasladan con velocidad constante. El segundo algoritmo implementado es una adaptación del reciente algoritmo de **Vasarhelyi**, definido por las siguientes reglas:

- **Repulsión para evitar colisiones entre agentes.**
- **Alineación de la velocidad para dirigir a los agentes hacia una dirección media.**
- **Autopropulsión para ajustarse a un valor de velocidad preferido.**

Además de estas reglas, el algoritmo incluye fuerzas de fricción que reducen las oscilaciones y facilitan la implementación en robots reales.

Ambos algoritmos proponen un comportamiento de evitación de obstáculos para hacer frente a los obstáculos convexos. En varias aplicaciones de ingeniería (por ejemplo, cartografía, cobertura de áreas, búsqueda y rescate), se requiere que el enjambre vuele en una dirección específica. Para ello, se permite la selección entre el consenso sobre la velocidad en el algoritmo de **Olfati-Saber**, o la

alineación de la velocidad en el algoritmo de **Vasarhelyi** y un llamado término de migración que penaliza las desviaciones de una velocidad determinada.

SwarmLab ofrece dos modalidades para la simulación de enjambres: el modo de alta fidelidad simula drones cuadricópteros, donde se implementa una dinámica y un control realistas, mientras que el segundo aproxima la dinámica de los drones con la dinámica de una masa puntual, cuyo estado se define por la posición inercial y la velocidad. El objetivo es compensar la fidelidad de la simulación y la eficiencia computacional.

Además de las clases y métodos que contiene este software implementados se ha introducido también diversas formas de interactuar con los objetos. Una de ellas es el uso de **Interfaces Gráficas** (GUI), actualmente existen dos interfaces gráficas, una para seleccionar los parámetros relacionados con simulaciones de un único dron y otro para simulaciones multi-dron o *swarms*. Por otro lado se encuentra el uso de **herramientas de visualización** para verificar y validar el código implementado, junto al análisis del estado del sistema; permitiendo realizar el seguimiento de los agentes, la velocidad del conjunto, la aceleración,...

Para finalizar el flujo de trabajo de una simulación de enjambre se encuentra resumido en la Figura 19. Para iniciar una simulación basta con ejecutar un *script* de ejemplo o interactuando con una interfaz gráfica de usuario (GUI) incluida en el software de *SwarmLab*. Cuando se inicia la simulación, se crean varias instancias de *drones* y se añaden al enjambre (*swarm*). Además, es posible instalar un visor de enjambre para visualizar la evolución del estado del enjambre durante el tiempo de simulación. El bucle principal de simulación calcula en cada iteración los comandos de control para cada dron del enjambre y actualiza sus estados. Los comandos de control para un dron dado sólo dependen de sus drones vecinos. Dependiendo de la elección del usuario, la clase *Swarm* utiliza un algoritmo de enjambre diferente para calcular los comandos de cada dron. Alternativamente, el usuario puede implementar y probar su propio algoritmo de control como un método de la clase *Swarm* donde los estados de los drones son accesibles, siguiendo los ejemplos disponibles.

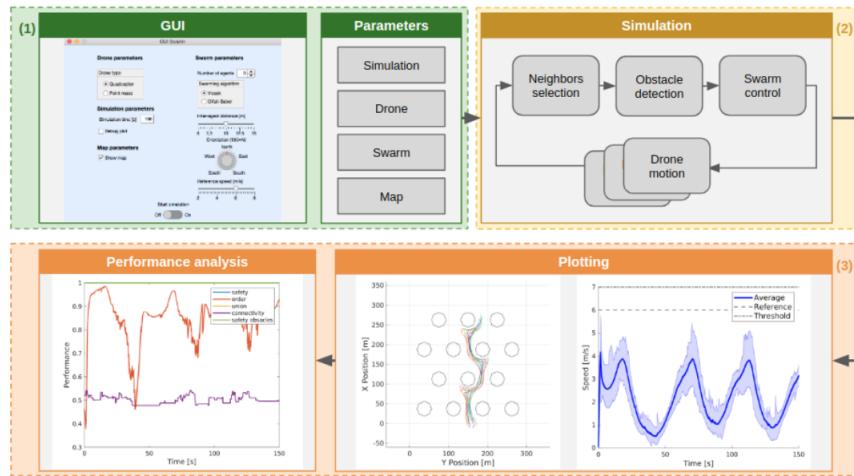


Figura 19: Flujo de trabajo de la aplicación

3.3. Descripción práctica

Entre los softwares analizados para la realización de pruebas prácticas de simulación de robótica de enjambre se encuentran diversas implementaciones que utilizan *ROS Kinetic* y *Gazebo*; como es el caso del *Swarm Gazebo Simulator* o del *RoboBees Colony Swarm Simulator*. No obstante, *SwarmLab* ofrece simulaciones en *Matlab* un software altamente utilizado en el sector académico, por lo que esta opción fue la seleccionada.

Para poder utilizar esta herramienta basta con tener instalada la aplicación de MATLAB junto a la *toolbox* de Machine Learning, mostrada en la Figura 20.

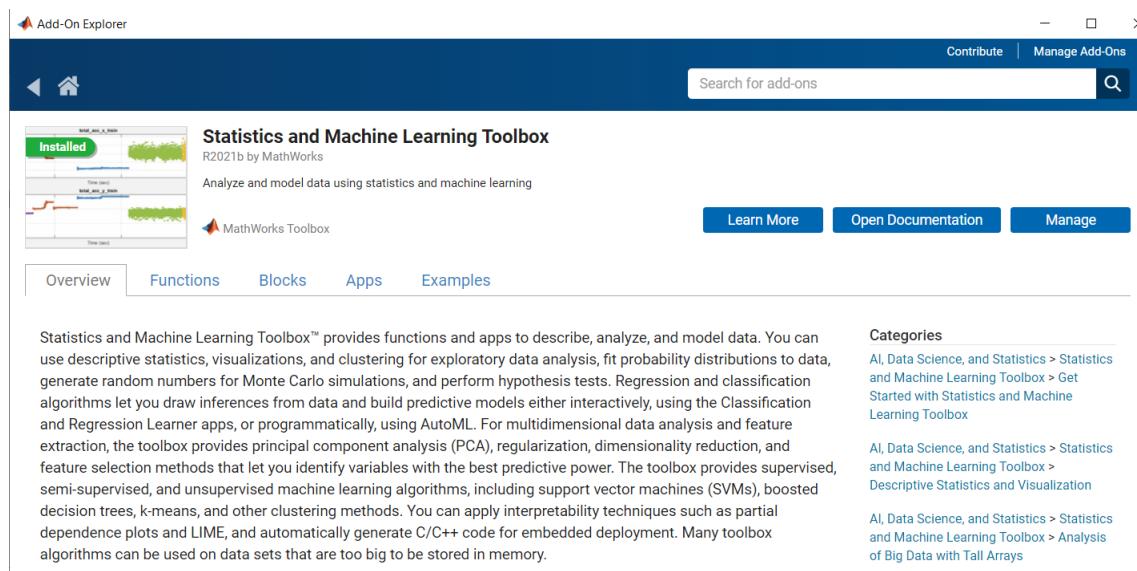


Figura 20: Machine Learning Toolbox.

Seguidamente es preciso descargar el repositorio de github de la librería de *SwarmLab*. Si se dispone de Git instalado se puede ejecutar el comando `git clone https://github.com/lis-epfl/swarmlab.git`.

El repositorio descargado debería ser similar al mostrado en la Figura 21.

Durante este caso de estudio han sido analizados los ejemplos que se encuentran en la carpeta de *examples*, dichos ejemplos se encuentran controlados únicamente por las interfaces gráficas desarrolladas, denominadas como *GUIL_drone.mlapp* y *GUIL_swarm.mlapp*. Por lo que estas aplicaciones serán las que serán utilizadas para la experimentación de este proyecto.

📁 @Drone	08/06/2021 09:39	File folder
📁 @Swarm	08/06/2021 09:39	File folder
📁 control	08/06/2021 09:39	File folder
📁 docs	08/06/2021 09:39	File folder
📁 enumerators	08/06/2021 09:39	File folder
📁 estimation	08/06/2021 09:39	File folder
📁 examples	14/01/2022 12:04	File folder
📁 graphics	08/06/2021 09:39	File folder
📁 guidance	08/06/2021 09:39	File folder
📁 math_tools	08/06/2021 09:39	File folder
📁 parameters	08/06/2021 09:39	File folder
📁 physics	08/06/2021 09:39	File folder
📁 tests	08/06/2021 09:39	File folder
📁 turbulent_wind_generator	08/06/2021 09:39	File folder
📄 .gitignore	08/06/2021 09:39	Text Document
📄 LICENSE	08/06/2021 09:39	File
⬇ README.md	08/06/2021 09:39	Markdown Source ...

Figura 21: Repositorio.

3.3.1. GUI_drone

Tras iniciar la ejecución de la aplicación de *GUI_drone.mlapp* se puede visualizar la interfaz gráfica mostrada en la Figura 22.

En la interfaz gráfica mostrada es posible seleccionar el tipo de dron se desea simular, sin embargo, los drones de tipo *Fixed-wing* no se pueden simular, puesto que sin importar la configuración de la simulación, al iniciar la misma se obtiene un error para realizarla. En la columna del medio es posible modificar los parámetros de la simulación, la primera fila modifica el modelo de orientación del mismo, siendo posible seleccionar entre control manual del mismo, seguimiento de una trayectoria, un gestor de trayectorias y un generador de trayectorias. Donde los tipos de trayectorias posibles para el gestor y seguidor de trayectorias son *Fillet* y *Dubin*, mientras que para el generador de trayectorias la forma de generar dicha trayectoria es *StraightRRT*. También existe una casilla en la que es posible de mostrar las gráficas de debug del mismo, que como se puede visualizar en la figura 23, contiene datos de la posición, velocidad, aceleración, el aire, los actuadores,... No obstante, esta funcionalidad solamente funciona para el controlador y el seguidor de trayectorias (*Controller* y *Path follower*).

En la columna de la derecha es posible modificar los parámetros del mapa, mostrándolo u ocultándolo e incluyendo viento o ráfagas. Sin embargo estas opciones únicamente se encuentran activas a la vez en el modo de controlador, siendo posible activar solamente las ráfagas en el resto de tipos de control. Finalmente, para iniciar la simulación es preciso posicionar el switch de la sección de *Start Simulation* en On.

En la figura 24 se muestran los distintos modos de funcionamiento.

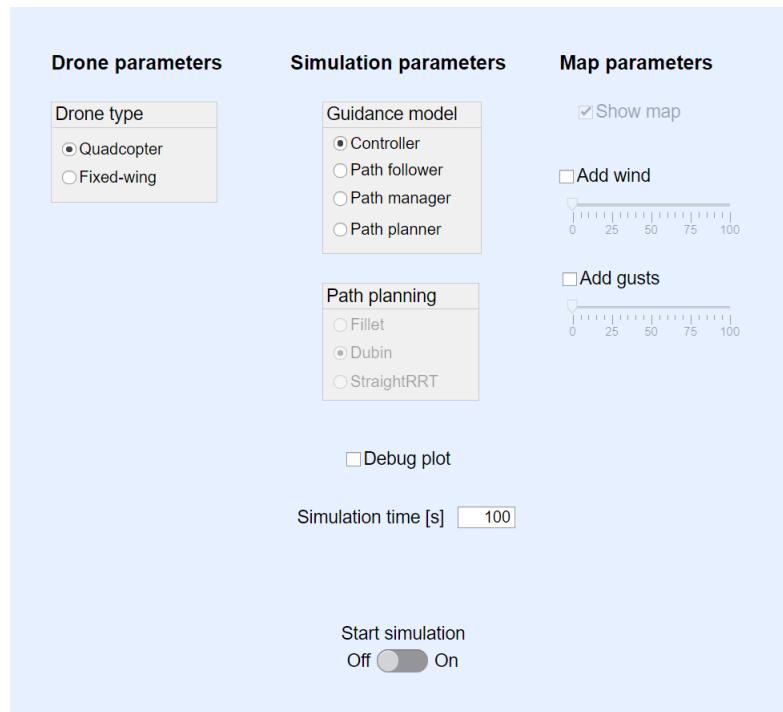


Figura 22: Interfaz gráfica para simular un dron

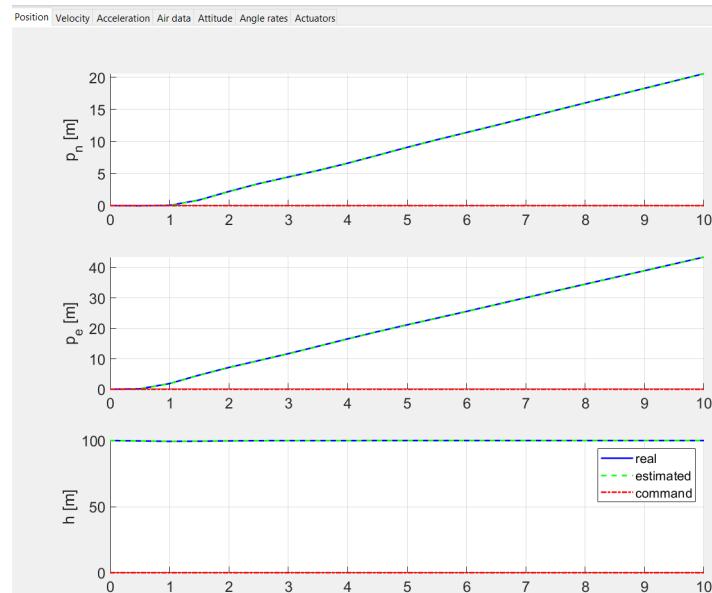


Figura 23: Debug Window.

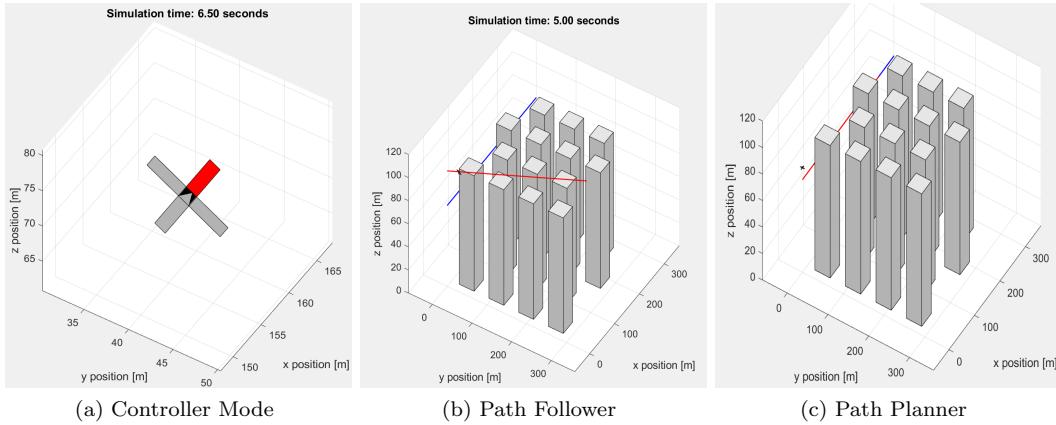


Figura 24: Modos de funcionamiento

3.3.2. GUI_swarm

Tras iniciar la ejecución de la aplicación de *GUI_swarm.mapp* se puede visualizar la interfaz gráfica mostrada en la Figura 25.

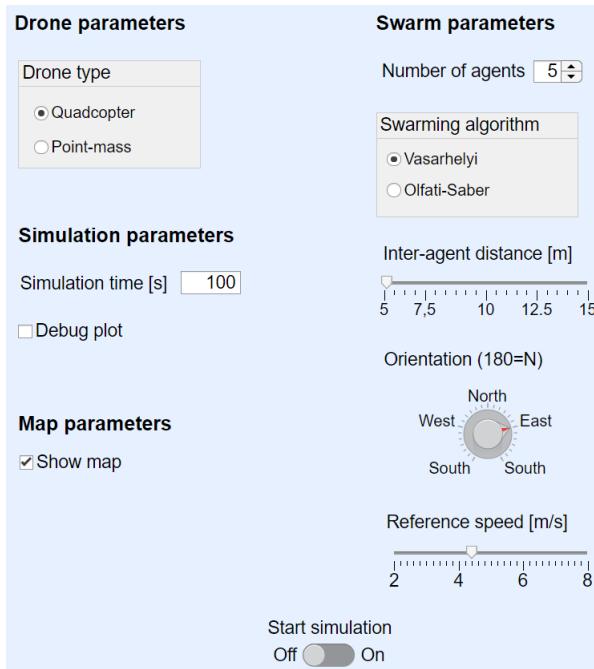


Figura 25: Interfaz de control de Swarm

Donde es posible configurar el tipo de dron que se desea simular, en este caso sí que funcionan correctamente ambos tipos de representación, sin embargo es recomendable que se utilice el método

de representación de punto de masa. Como en la interfaz de usuario anterior también es posible de mostrar las gráficas de debug como se muestra en la Figura 26. Además se puede modificar el número de agentes, o drones, el algoritmo utilizado para el control del enjambre, la distancia entre agentes, hacia dónde se desplazan estos y la velocidad de referencia de los mismos.

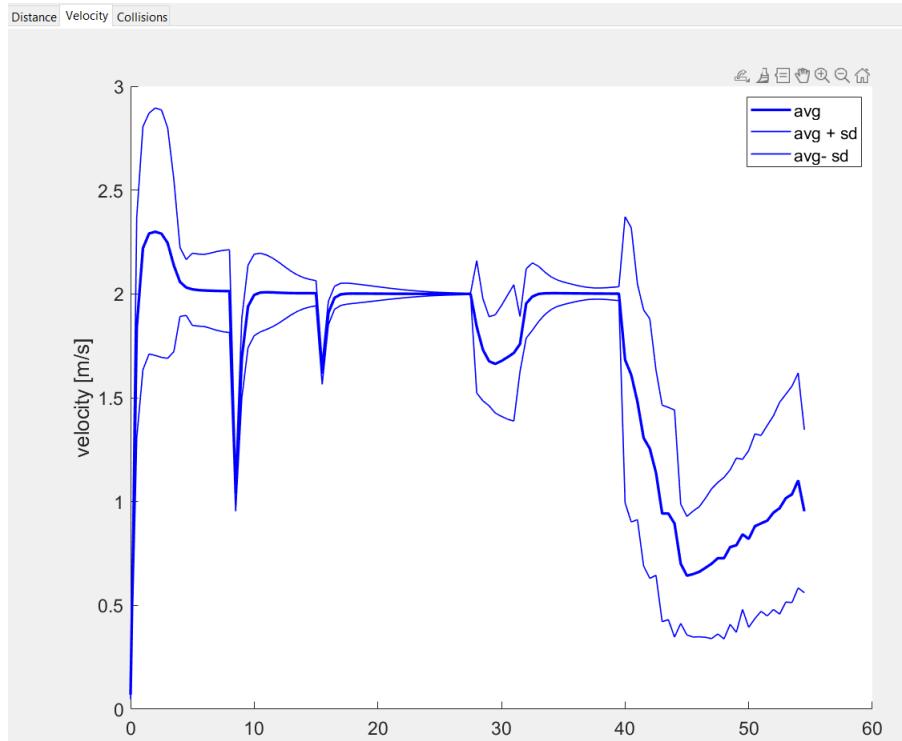


Figura 26: Gráficas de debug del modo Swarm

Una vez se ejecuta la simulación es posible controlar el enjambre mediante la modificación de la distancia entre agentes, la orientación y la velocidad de estos. Una vez finalizada la simulación se guardará una grabación de la misma en la carpeta de *results*, junto a los resultados de la simulación de los drones individuales. La reproducción de la grabación de la simulación se inicia automáticamente una vez transcurrido el tiempo de simulación determinado inicialmente. Durante dicha reproducción es posible visualizar la trayectoria recorrida por cada individuo, un ejemplo de dicha reproducción se representa en la Figura 27.

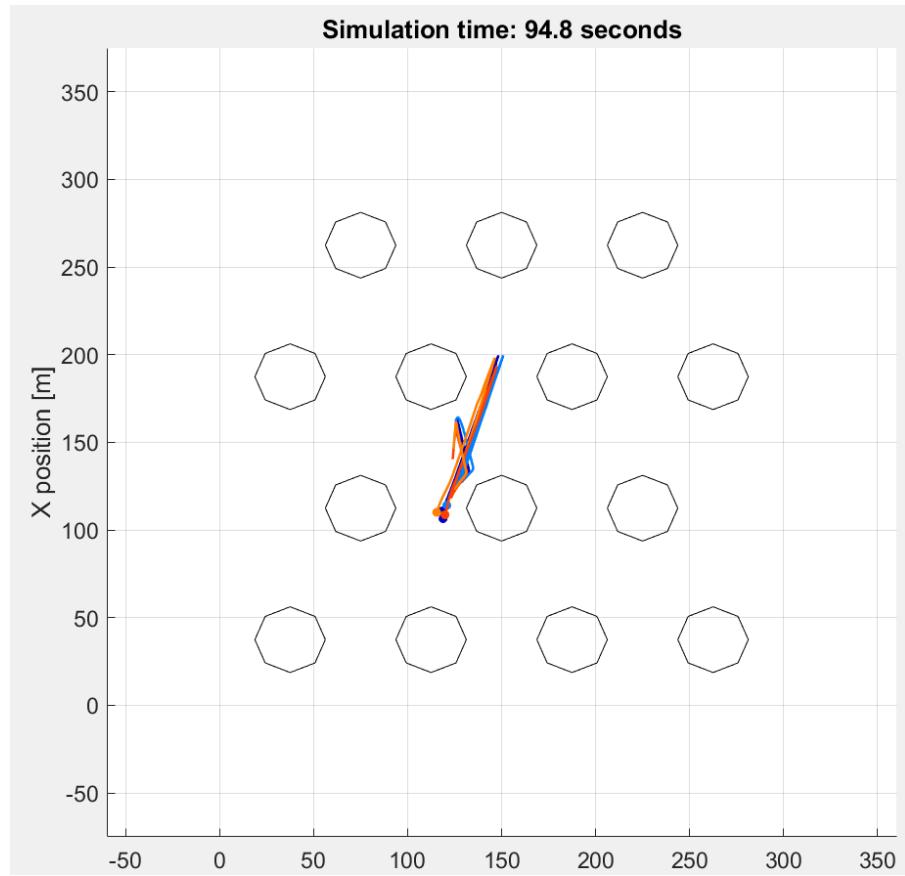


Figura 27: Reproducción de la simulación realizada

Además de esta reproducción de la simulación se muestra una serie de gráficas, también almacenadas en la carpeta de *results* que muestran la distancia entre cada agente del enjambre durante toda la simulación (Figura 28a), la velocidad media e instantánea de los mismos durante la simulación (Figura 28b) y la aceleración de estos (Figura 28c).

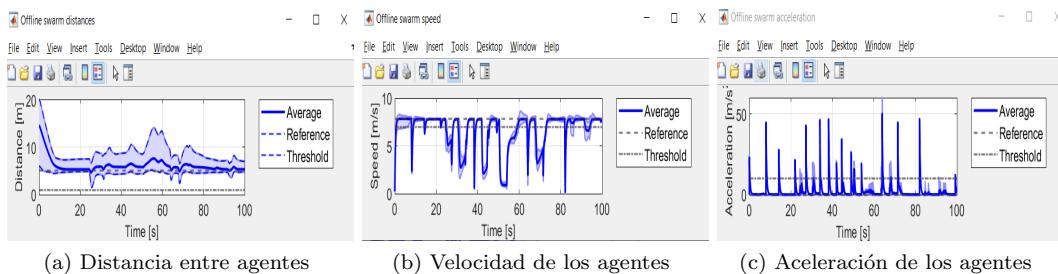


Figura 28: Resultados de la simulación de los agentes.

Para finalizar, se obtienen dos gráficas finales, las cuales muestran la distancia media entre el conjunto y los obstáculos del escenario durante cada instante de la simulación, mostrado en la figura 29, además de una gráfica en la que se puede visualizar un análisis del desarrollo de la simulación determinando el orden, la unión, la conectividad y la seguridad de estos durante la simulación, esto se muestra en la Figura 30.

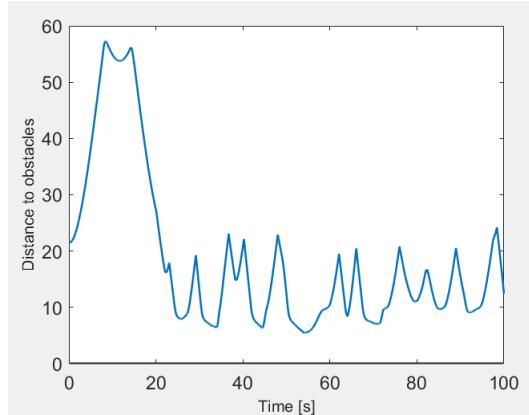


Figura 29: Distancia entre el conjunto y los obstáculos durante la simulación.

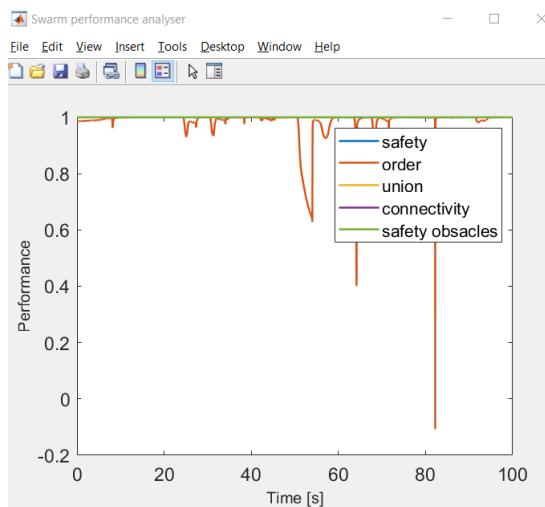


Figura 30: Desarrollo del enjambre durante la simulación.

4. Conclusiones

La robótica de enjambres parece ser un enfoque prometedor cuando se deben realizar diferentes actividades al mismo tiempo, cuando se desea una alta redundancia y la ausencia de un único punto de falla, y cuando es técnicamente inviable configurar la infraestructura necesaria para controlar los robots de forma centralizada. También es eficiente en diversas tareas como se vio anteriormente, sobre todo en aquellas que precisan de comunicación entre los robots para abarcar un entorno amplio en el cual realizar una tarea conjunta. No obstante, este es aún un campo de la robótica que se encuentra en exploración debido al alto coste que requiere el diseño de algoritmos efectivos que engloben al enjambre, en un futuro probablemente este tipo de problemas se solucionen y se logre explotar este avanzado campo de la robótica que ofrece flexibilidad y adaptabilidad a multitud de tareas.

En cuanto lo que respecta al caso de estudio analizado es posible destacar que la idea inicial es bastante acertada, puesto que un usuario que desee iniciarse en el ámbito de la robótica en enjambre aérea no tiene por qué estar familiarizado con ROS, Gazebo, Ardupilot u otros tipos de simuladores que requieren un determinado conocimiento de distintos tipos de lenguajes de programación. Además, *MATLAB* es un entorno de programación altamente utilizado en el ámbito académico por las características del mismo. Sin embargo, la implementación no está completamente realizada, puesto que dentro del programa existen diversos errores que imposibilitan la correcta ejecución del sistema en diversos casos. Por lo que se podría concluir que este caso de estudio se encuentra enfocado a un usuario final bastante indicado, pero su falta de desarrollo o continuidad hace que este no pueda ser una alternativa completa al resto de simuladores de robótica de enjambre aéreo.

Referencias

- [1] Y. TAN, Z. ZHENG, *Research Advance in Swarm Robotics*, Defence Technology, 9(1), 18–39, (2013).
- [2] J. C. BARCA, Y. A. SEKERCIOGLU, *Swarm robotics reviewed*, Robotica, 31(03), 345–359, (2012).
- [3] L. BAYINDIR, *A review of swarm robotics tasks*, Neurocomputing, 172, 292–321, (2016).
- [4] M. BRAMBILLA, E. FERRANTE, M. BIRATTARI, M. DORIGO, *Swarm robotics: a review from the swarm engineering perspective*, Swarm Intelligence, 7(1), (2013).
- [5] G. BENI, *From Swarm Intelligence to Swarm Robotics*, Lecture Notes in Computer Science, 1–9, (2005).
- [6] E. ŞAHİN, S. GİRGİN, L. BAYINDIR, A. E. TURGUT, *Swarm Robotics*, Swarm Intelligence, 87–100.
- [7] I. NAVARRO , F. MATÍA, *An Introduction to Swarm Robotics*, (2012).
- [8] L. BAYINDIR, W. SAHIN, *A Review of Studies in Swarm Robotics*, Turk J Elec Engin, VOL.15, NO.2, (2007).
- [9] JAMES McLURKIN, JENNIFER SMITH, JAMES FRANKEL, DAVID SOTKOWITZ, DAVID BLAU, BRIAN SCHMIDT, *Speaking Swarmish: Human-Robot Interface Design for Large Swarms of Autonomous Mobile Robots*.
- [10] ALI EMRE TURGUT, HANDE ÇELIKKANAT, LEVENT BAYLNDIR, *Kobot: A mobile robot designed specifically for swarm robotics research*.
- [11] KERTH, G., : *Causes and consequences of sociality in bats*. BioScience, 58(8), 737–746 (2008).
- [12] GARCÍA-AUNON, P., ROLDÁN, J.J., DE LEÓN, J., DEL CERRO, J., BARRIENTOS, A., *Practical applications using multi-UAV systems and aerial robotic swarms*, Revista Iberoamericana de Automática e Informática Industrial, 18, 230-241, (2021).
- [13] ACUÑA CALVIÑO, D, *Tecnologías asociadas a sistemas de enjambres de μUAV*, Documentos de Seguridad y Defensa, 101-148, (2012).
- [14] ENRICA SORIA, FABRIZIO SCHIANO, DARIO FLOREANO, *SwarmLab: a Matlab Drone Swarm Simulator*
<https://github.com/lis-epfl/swarmlab>
- [15] NEVALSAR, *Swarm Gazebo Simulator*
<https://github.com/nevalsar/swarm-gazebo-simulator>
- [16] BRYANKATE, *RoboBees Colony Swarm Simulator*
<https://github.com/bryankate/simbeeotic>
- [17] ETHZ-ASL, *RotorS*
https://github.com/ethz-asl/rotors_simulator
- [18] R. OLFATI-SABER, *Flocking for Multi-Agent Dynamic Systems: Algorithms and Theory*, IEEE Transactions on Automatic Control, vol. 51, no. 3, pp. 401–420, 2006.

- [19] G. VAS ARHELYI, C. VIRAGH, G. SOMORJAI, T. NEPUSZ, A. E. EIBEN, AND T. VICSEK, *Optimized flocking of autonomous drones in confined environments*, Science Robotics, vol. 3, no. 20, 2018.