

Sistemas multirrobot: etapas de diseño, aplicaciones y problemas abiertos

La rápida expansión de robots autónomos a nuestro alrededor, no solo ha planteado la posibilidad de mejorar su desempeño, sino también de ampliar sus alcances, mediante la comunicación y colaboración entre múltiples robots. En este contexto, surgen los llamados “sistemas multirrobots” (SMR), los cuales presentan muchas características interesantes como la heterogeneidad de acciones disponibles, la tolerancia ante fallos y una mayor eficiencia. En este trabajo se realiza una revisión de los principales trabajos en la materia, abordando las distintas etapas de diseño y subtarefas requeridas para la ejecución de un multirrobot, como así también las aplicaciones más relevantes y los tests de desempeño asociados. Junto a ello, se identifican las líneas de investigación vigentes y los principales problemas abiertos como potenciales campos de trabajo.

**Juan Luis Rosendo,
Fernando Valenciaga y Fabricio Garelli**
Grupo de Control Aplicado
LEICI, Facultad de Ingeniería
Universidad Nacional de La Plata
<https://leici.ing.unlp.edu.ar/>

Los robots autónomos han ganado un espacio de aplicación tanto en el sector industrial como en nuestros hogares. Modelos comerciales para aplicaciones de almacenamiento y transporte, exploración, cartografía, agricultura de precisión, tareas domésticas, entre otras, ya se encuentran disponibles y auguran una creciente expansión del uso de estos dispositivos. Tomando esto en cuenta, resulta natural asumir que al aumentar el número de robots autónomos a nuestro alrededor, estos comenzarán a comunicarse y completar sus actividades en forma colaborativa.

La comunicación entre distintos robots autónomos ha dado lugar a los sistemas multirrobot (SMR). Estos sistemas posibilitan alcanzar objetivos globales que los robots individuales no podrían realizar. En estos sistemas, la posibilidad de reasignar tareas dinámicamente incrementa la flexibilidad de adaptación ante distintos escenarios. Asimismo, la multiplicidad de robots asegura una mejor tolerancia a fallos y una mayor eficiencia en la ejecución de tareas. Sin embargo la utilización y el control de los SMR tiene una complejidad inherente mayor que la de un robot individual: división de tareas, utilización de información distribuida, supervisión global, co-

Contactos

El artículo acá presentado fue elaborado en consideración de una amplia bibliografía sobre el tema. Para mayores detalles, el y la interesada pueden contactar directamente a los autores.

Juan Luis Rosendo, juanluis.rosendo@ing.unlp.edu.ar

Fernando Valenciaga, fval@ing.unlp.edu.ar

Fabricio Garelli, fabricio@ing.unlp.edu.ar

Nota del editor.

El artículo aquí publicado, parcialmente financiado por la Universidad Nacional de la Plata (I216,I253,I258) y CONICET (PIP0837), fue presentado originalmente como trabajo de investigación en el Congreso Argentino de Control Automático celebrado en 2020, en el marco de la Semana de Control Automático llevada adelante por AADECA.

municación entre robots, etc. son solo algunos de los problemas para abordar.

Los multirroboots son un caso particular de los llamados sistemas de múltiples agentes (MAS, por sus siglas en inglés), donde los agentes están restringidos a sistemas robóticos físicos. (En este trabajo, utilizaremos indistintamente los términos "agente" y "robot"). Los múltiples agentes se caracterizan por agentes que interactúan con objetivos comunes o en conflicto, trabajando con información descentralizada e incompleta, en forma asíncrona y bajo un control descentralizado. Debido a su estrecha relación es que gran parte de la investigación llevada adelante para los múltiples agentes ha sido extendida a los multirroboots.

Este trabajo pretende servir de breve introducción al campo de los SMR. Se espera que el lector obtenga un panorama general de estos sistemas, sus desarrollos actuales y sus potenciales aplicaciones. La principal diferencia entre este trabajo y otros trabajos de revisión previos radica, además de su carácter introductorio, en su enfoque abarcativo de las distintas etapas y áreas de diseño en la ejecución de los multirroboots y en la clasificación de los trabajos por tipo de aplicación y test de desempeño.

El trabajo se organiza y presenta de la siguiente manera. La sección siguiente presenta una descripción de la estructura y etapas de trabajo de los multirroboots. Luego, se ofrece un análisis de las tareas colaborativas y aplicaciones más relevantes; a continuación, las principales líneas de investigación actuales y problemas abiertos. Finalmente, las conclusiones.

La multiplicidad de robots asegura una mejor tolerancia a fallos y una mayor eficiencia en la ejecución de tareas

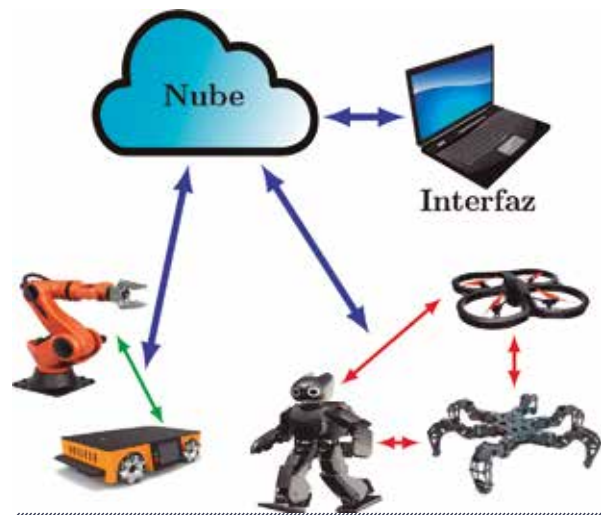


Figura 1. Estructura típica de un multirrobot

Estructura y esquema de trabajo en multirroboots

Dentro de la bibliografía especializada, existen varios artículos que abordan un estudio genérico y la taxonomía de los multirroboots y sus aplicaciones. Los criterios empleados para este propósito se vinculan con el grado de diversidad y nivel de comunicación de los agentes, con la diversidad de actuadores/sensores de cada agente, el tipo de interacción y los tipos de tareas realizadas. En este trabajo, se realiza una reelaboración, combinación y síntesis de los conceptos encontrados en contribuciones previas.

Estructura de un multirrobot

Los sistemas multirrobot genéricos estructuran sus componentes en una forma jerárquica como la que se presenta esquemáticamente en la figura 1. Habitualmente, en el nivel de mayor jerarquía se encuentran los procesos que se ejecutan en la nube o servidor. En este ámbito, se encuentra definida la tarea global y la información de todos los agentes. Estos últimos pueden, tanto

acceder a la información allí disponible, como también emplear su capacidad de cómputo para fines particulares. Al igual que los agentes, la interfaz de control se puede conectar al servidor o bien encontrarse allí mismo.

En un segundo nivel jerárquico, se encuentran los subconjuntos de robots (coaliciones o flotas), a los cuales se les asignan subtareas de la tarea global. Los integrantes de estos subconjuntos se pueden comunicar entre sí a un nivel local. Los tipos de interacción que pueden exhibir son cooperativa, competitiva o colaborativa basada en objetivos o recursos y habilidades de cada agente.

Habitualmente, en el nivel de mayor jerarquía se encuentran los procesos que se ejecutan en la nube o servidor.

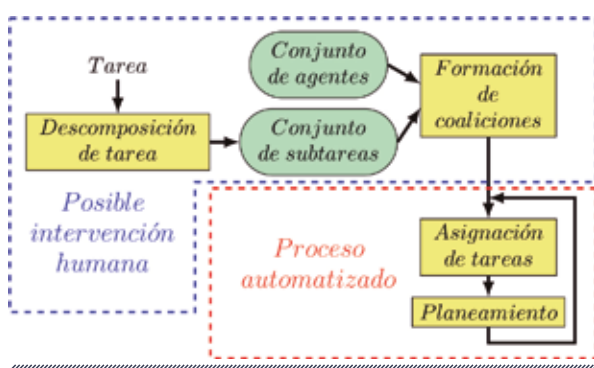


Figura 2. Etapas de trabajo en un multirrobot

El nivel de jerarquía más bajo es el que corresponde a los agentes individuales. Estos pueden acceder a sus sensores, controlar sus actuadores y conectarse tanto a la nube como a sus compañeros de flota.

En particular, en los multirrobots heterogéneos, los agentes pueden tener distintas arquitecturas cognitivas (reactiva, deliberativa o híbrida) o diferentes características físicas (robots terrestres, aéreos o marinos, por ejemplo).

» Etapas de trabajo y áreas de diseño en los multirrobots

Independientemente de las características estructurales de los multirrobots, su funcionamiento como sistema se basa en la resolución de cuatro subtareas (ver figura 2). Actualmente, los principales estudios y aportes dentro del campo de los multirrobots se realizan dentro de cada uno de estos bloques.

El primer paso en el flujo de trabajo de los multirrobots consiste en tomar una tarea compleja (i.e. una tarea que sobrepasa las posibilidades de un robot individual) y dividirla en subtareas independientes o secuencialmente dependientes.

» Descomposición de tareas

El primer paso en el flujo de trabajo de los multirrobots consiste en tomar una tarea compleja (i.e. una tarea que sobrepasa las posibilidades de un robot individual) y dividirla en subtareas independientes o secuencialmente dependientes. Esta división y asignación puede o no ser automatizada. Algunas propuestas dejan esta responsabilidad en manos del operador. En los enfoques automáticos se observa que la des-

composición de una tarea se asocia a la asignación de los agentes. Esto produce una realimentación continua que, en función del ambiente de trabajo dinámico, resuelve el problema de asignación iterativamente.

» Formación de coalición

Esta etapa divide los agentes en grupos o equipos que pueden ser cooperativos o no [20]. La formación de equipos puede realizarse de forma offline y estática, o de forma dinámica online ajustando los equipos al espacio de trabajo y las tareas que se desea desarrollar. El problema para resolver consiste en cómo seleccionar el mejor grupo de agentes para resolver una determinada tarea. En este sentido, se puede encontrar una gran cantidad de trabajos que presentan enfoques con algoritmos de búsqueda tales como *colony optimization*, *particle swarm optimization* y diversos algoritmos genéticos. Los algoritmos de búsqueda se caracterizan por ser simples, aunque su rendimiento depende de la métrica utilizada para evaluar a los agentes y de su compatibilidad. Otros enfoques utilizan métodos de subasta o de aprendizaje por refuerzo (RL, por sus siglas en inglés), donde cada agente aprende de sus vecinos en función de las interacciones realizadas.

» Asignación de tareas

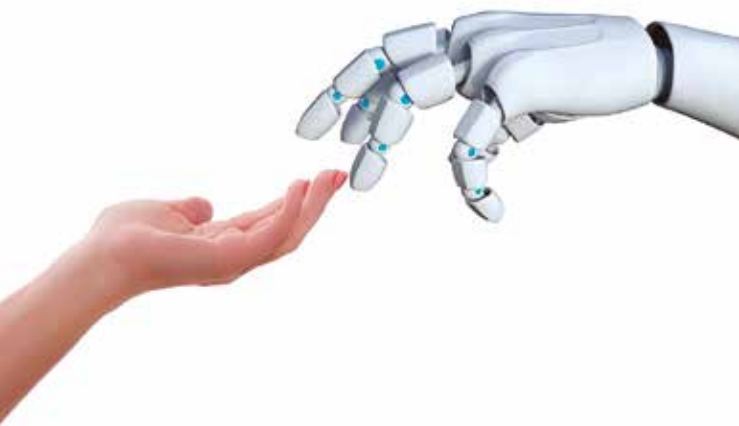
Esta etapa busca asignar a un robot o grupo de robots las diferentes sub tareas específicas definidas, y se puede implementar en forma centralizada o descentralizada. El objetivo es lograr una asignación óptima con los agentes disponibles. Es posible encontrar dentro de la bibliografía disponible varios trabajos que recopilan métodos de asignación. Estos métodos deben operar bajo restricciones espaciales, temporales, de limitación de sensores o actuadores, y a su vez deben considerar la interrelación entre las sub tareas. El tipo de algoritmo más extendido para la asignación

es el denominado “de mercado” o “de subastas”. En él, el algoritmo publica en la red del multirrobot una tarea determinada, y cada uno de los agentes ofrece su costo para realizarla (posición, energía disponible, costo, etc.). Luego, a partir de cierto criterio definido, se selecciona el más conveniente para la operación. Debe destacarse que la asignación de tareas se encuentra interrelacionada con la mejor elección de robots para su ejecución. Esta particularidad ha sido abordada en varios trabajos donde ambas tareas se realizan en simultáneo.

» Planeamiento distribuido

En esta etapa se determina la secuencia de acciones que debe llevar a cabo cada uno de los agentes con el propósito de completar las sub tareas asignadas. El grupo de algoritmos y modelos que realizan este tipo de determinaciones son conocidos como “algoritmos de toma de decisión” y, en general, basan sus evaluaciones en base a políticas de optimización de tiempo, espacio y complejidad. En este campo se puede encontrar una gran variedad de propuestas. Entre las más extendidas, se encuentran las siguientes:

- Inteligencia de enjambre. Inspirada en el comportamiento social de ciertos animales, este tipo de desarrollos imitan el comportamiento de varios agentes cooperativos autónomos. En particular, estos modelos se caracterizan por ser autoorganizados con una estructura descentralizada donde los agentes se comunican solo localmente.
- Modelos basados en teoría de juegos. Los agentes poseen incentivos (funciones de utilidad) y leyes de control adaptativas (reglas de aprendizaje). Estas características inducen un comportamiento colectivo deseable a través de interacciones locales que respetan la arquitectura de decisión distribuida.



- Aprendizaje por refuerzo. Permite a los agentes aprender de sus acciones mediante su recompensa o castigo. Esta política de recompensas puede ser distribuida o global al sistema.

Aplicaciones y pruebas de desempeño

Existe un sinnúmero de aplicaciones en las que se pueden utilizar multirrobo. Estas suelen asociarse a pruebas piloto que se realizan sobre los multirrobo a fin de evaluar su coordinación y rendimiento. A continuación, se listan las pruebas de desempeño más difundidas, remarcando aplicaciones reales que comparten requerimientos similares:

- » Búsqueda de alimento y cobertura. La tarea de búsqueda consiste en que los agentes inicien la exploración del entorno de trabajo en busca de determinados objetos sin ubicación previa determinada. Una vez que estos objetos son encontrados, deben ser llevados a puntos de depósito. Uno de los principales problemas que involucra esta tarea es lograr que los robots actúen sin interferirse mutuamente cuando cubren el área designada. El

caso particular donde los agentes del sistema deben desplegarse en el ambiente de forma de cubrirlo completamente en forma cooperativa y óptima, es conocido como problema de cobertura. Casos típicos de estas aplicaciones son robots de limpieza, búsqueda y rescate, remoción de minas, remoción de nieve, trabajos de pintura, etc.

- » Observación de múltiples objetivos. En este caso, la tarea consiste en que un grupo de robots detecte y siga a un grupo de objetos móviles. Si bien el problema de detección ha sido abordado previamente por redes de sensores distribuidos, aplicaciones más ambiciosas potencian la capacidad de estas redes con la utilización de robots móviles. Los usos típicos están relacionados con tareas de vigilancia y reconocimiento.
- » Transporte de objetos. Existen típicamente dos pruebas de este tipo: a) la primera de ellas, conocida como "box pushing", consiste en empujar objetos a una determinada configuración final, usualmente en un plano. Se la asocia a tareas de almacenaje, carga y descarga de camiones, etc.; b) la segunda, si bien es similar a la anterior, considera espacios de tres dimensiones, incluyendo operaciones verticales de levantamiento y descenso de objetos. Estas aplicaciones involucran objetos de grandes dimensiones que superan la capacidad de un robot individual. Ejemplos de este tipo implican transporte y ensamblaje, construcción, etc. La dificultad de estas pruebas radica en la coordinación del movimiento y el control de formación.
- » Exploración y movimiento coordinado (flocking). En las tareas de exploración, los robots se dispersan en el entorno de trabajo con el objetivo de recolectar la mayor cantidad de información posible. En las tareas de movimiento coordinado, los robots se mueven como una manada. El foco está puesto en que los robots puedan mantener una formación. Las principales complicaciones en

este tipo de tareas es la ubicación de cada robot en el ambiente, así como la fusión de información recolectada individualmente. Estas operaciones se asocian a las aplicaciones de transporte en puertos y patios de maniobras, exploración de zonas peligrosas, agricultura de precisión, autoensamblado, etc. Un subgrupo de aplicaciones incluidas en esta categoría son aquellas que buscan la elaboración de mapas del ambiente de trabajo, buscando resolver el problema de localización y mapeo (SLAM) en forma cooperativa.

- » Competencia de equipo. En los últimos tiempos han surgido diferentes competencias que hacen foco en la implementación de multirroboots. Es habitual que en ellas se propongan diferentes combinaciones de las pruebas enumeradas anteriormente, buscando someterlos a distintas actividades con dinámicas inciertas y ambientes desconocidos u hostiles. Algunas de estas competencias buscan la coordinación de multirroboots heterogéneos en tareas de búsqueda y rescate o competencias de coordinación como el fútbol robótico.

Existe un sinnúmero de aplicaciones en las que se pueden utilizar multirroboots.

Problemas abiertos

De las principales áreas de investigación y las aplicaciones presentadas, se observa que el grado de maduración en la tecnología de los multirroboots es intermedio. Aún quedan problemáticas importantes para resolver y muchas potenciales mejoras por desarrollar. A continuación, se enumera un posible conjunto de tópicos de I+D para abordar:

- » Coaliciones dinámicas. En los multirroboots en donde sus miembros no se conocen entre sí en la etapa de diseño, los robots deben formar coaliciones temporales a fin de lograr los objetivos. Un ejemplo de esta situación se puede considerar en la intersección de dos caminos con vehículos autónomos; en este caso, la formación de la coalición es continua y dura poco tiempo (lo que le lleve a cada vehículo atravesar la intersección). Este tipo de aplicación implica que los agentes involucrados reconozcan su participación en la coalición, “sepan” quiénes conforman la coalición, intercambien información, y a su vez consideren la seguridad de la red formada. Aunque este es un problema abierto, la comunicación basada en atributos es una posible solución. Esta es una variante de la comunicación publicador/suscriptor, pero donde los potenciales receptores son discriminados por los contenidos del mensaje, posibilitando la comunicación con agentes anónimos.

Se observa que el grado de maduración en la tecnología de los multirroboots es intermedio. Aún quedan problemáticas importantes para resolver

- » Heterogeneidad vs. escalabilidad. Una de las principales características de los multirroboots es la capacidad de tener agentes heterogéneos que le permiten adaptarse a ambientes cambiantes. Sin embargo, esa misma ventaja choca con la potencial escalabilidad del sistema. Aunque en la literatura sobre el tema se proponen varios algoritmos de control descentralizados, el balance entre heterogeneidad y escalabilidad del sistema continúa siendo un desafío abierto. Enfoques jerárquicos distribuidos donde la formación

de subgrupos permita interacciones locales fluidas y globales más reducidas puede ser una solución a esta problemática.

- » Conocimiento compartido. En distintos desarrollos de multirrobo, el modelado del entorno se forma a partir de la información recolectada y compartida por el grupo de robots. Si este modelo compartido se consulta frecuentemente, los recursos de red del sistema pueden resultar muy demandados, deteriorando críticamente el rendimiento del sistema. Otros problemas asociados a este contexto son la integración de datos heterogéneos, la garantía de consistencia en los datos y la escalabilidad. Soluciones basadas en el empleo de recursos de la nube y de almacenamiento distribuido se ven como aproximaciones parciales a la solución de este tipo de problema.
- » Interfaz hombre-robots. Las aplicaciones en donde los multirrobo no son totalmente autónomos, o bien aquellos conocidos como "human in the loop", que implican la interacción continua del usuario con los robots, requieren que el usuario sea capaz de interactuar y acceder a la información relevante del sistema para comandarlo en forma eficiente. La conciencia de situación y la carga de trabajo en el usuario han sido las temáticas más estudiadas en relación a este punto. Recientes estudios muestran que la tendencia para las futuras interfaces es la utilización de interacciones multimodales, tecnologías inmersivas como la realidad virtual y aumentada, y algoritmos adaptativos para facilitar el trabajo de los operadores.
- » Aprendizaje automático. La aplicación de técnicas de aprendizaje automático, en especial aquellas que requieren poca o ninguna intervención humana para la sintonización del algoritmo de aprendizaje, pueden permitir a los agentes de los multirrobo lidiar en una forma más eficiente con ambientes dinámicos. Aproximaciones utilizan-

do optimización bayesiana y redes neuronales aleatorias son posibles líneas de investigación abiertas.

Recientes estudios muestran que la tendencia para las futuras interfaces es la utilización de interacciones multimodales, tecnologías inmersivas como la realidad virtual y aumentada, y algoritmos adaptativos para facilitar el trabajo de los operadores.

Conclusiones

En este trabajo se revisaron las principales características de los multirrobo y las distintas etapas en su flujo de trabajo. A su vez, se presentaron las principales aplicaciones en que se ven involucrados hasta la fecha, los test o pruebas de desempeño para cada aplicación y los trabajos más relevantes asociados a ellas. La tecnología de los multirrobo se encuentra aún en fases iniciales de desarrollo. La integración de estos sistemas al concepto de IoT, la utilización de algoritmos que aprovechen los datos masivos, y los algoritmos de inteligencia artificial se presentan como las principales ideas y desafíos hacia el futuro de este tipo de sistemas. ■