

Benemérita Universidad Autónoma de Puebla

Facultad de Ciencias de la Computación



Una Arquitectura para la Coordinación de Múltiples Robots

Tesis presentada para obtener el título de:
Maestría en Ciencias de la Computación

Presenta: Mario Alberto Posada Zamora

Asesor: Dr. Abraham Sánchez López

Agosto 2017

RESUMEN

En el área extensa de la robótica se han abordado tareas que se han vuelto cotidianas como la exploración de entornos desconocidos, el mapeo de espacios, el ensamblaje de piezas industriales, entre otras tareas.

Independientemente del problema a resolver, uno de los aspectos importantes y poco explorados en la robótica es una sub área de la misma llamada robótica colectiva. La robótica colectiva aborda tareas cotidianas de la robótica común, sin embargo, la resolución de dichas tareas es llevada a cabo por más de un agente robótico trabajando en el mismo plano espacial y alcanzando un objetivo general en común.

Sin embargo, con la introducción de más de un agente robótico trabajando en un entorno, es necesario tener mecanismos de comunicación y de coordinación para asegurar que el objetivo en común sea alcanzado minimizando, según sea posible, las interferencias causadas por la presencia y acciones de todos los agentes interactivos.

Los principios de la robótica colectiva obedecen a muchas normas que se encuentran en las formas comunes de interacción social humana como de algunos comportamientos grupales observados en otros seres vivos.

Esencialmente, los dos problemas de colaboración antes mencionados, la coordinación y la comunicación, son los que deben establecerse de acuerdo al tipo de agentes robóticos con los que se cuente, por lo que, si un sistema compuesto de múltiples agentes robóticos posee robots de diversas especies y características, sería un desafío lograr adaptar la colaboración tomando en cuenta las capacidades individuales de cada agente.

En el presente trabajo se propone el diseño de una arquitectura que coordina múltiples robots con diversas capacidades individuales de manera que puedan lograr tareas simples comunes, además de con ello poder destacar los problemas que se presentan al utilizar agentes robóticos heterogéneos (de diversas especies y diferentes características) con respecto a un sistema de agentes robóticos homogéneos (mismo tipo de robot).

La estructura del documento está formada por seis capítulos. En el primer capítulo se introduce el problema de la interacción de múltiples agentes en un mismo entorno, además de algunas propuestas que se han realizado con respecto a la resolución de la colaboración entre múltiples robots.

En el segundo capítulo se tratan todos los temas a profundidad sobre los sistemas multi robot en general, y los tipos de sistemas que existen, de acuerdo a toda la teoría que se hereda de los sistemas multi agente.

El tercer capítulo describe los elementos que deben conformar una arquitectura para coordinar múltiples robots, listando las opciones más comunes y las ventajas y desventajas de cada una de ellas.

El cuarto capítulo describe de manera detallada la propuesta específica y las justificaciones respectivas de cada elemento elegido dentro de la arquitectura. De igual manera se mencionan y describen las herramientas utilizadas.

El quinto capítulo introduce la tarea a resolver junto con las técnicas de resolución de la misma. Además, reporta las pruebas realizadas con la arquitectura con respecto a dicha tarea, para ver si el objetivo es alcanzado y las condiciones para cada prueba.

Por último, el sexto capítulo presenta al lector las conclusiones del presente trabajo, además de las observaciones encontradas en la realización del mismo. Dentro de las conclusiones se discute el papel importante que tuvieron las herramientas utilizadas en la realización del proyecto.

ÍNDICE GENERAL

Resumen	I
Índice general.....	III
Índice de figuras.....	VI
Índice de tablas.....	VIII
1 Introducción	1
1.1 Grupos de robots	2
1.2 Problemas de control de grupos de robots	3
1.3 Estado del arte	4
1.3.1 CEBOT.....	4
1.3.2 Sistema multi agente de vigilancia	7
1.3.3 ACTRESS	8
1.4 Justificación y contribuciones del trabajo	10
1.5 Organización del escrito de tesis.....	11
2 Sistemas multi robot y sistemas multi agente	13
2.1 De un solo robot al conjunto de robots	13
2.2 Agentes	14
2.3 Sistemas multi agente	16
2.4 Control cooperativo	17
2.5 Enfoques en los sistemas multi robot	20
2.5.1 Robótica basada en comportamientos.....	20
2.5.2 Robótica colectiva.....	21
2.5.3 Robótica Evolutiva	23
2.6 Problemas inherentes en un sistema multi robot.....	25
3 Arquitecturas para la coordinación multi robot	27
3.1 Protocolos y modelos de comunicación para la coordinación.....	29
3.1.1 Rango de comunicación limitado.....	32
3.1.1.1 Comunicación directa (Sistemas mensaje-diálogo)	33
3.1.2 Rango de comunicación ilimitado	34
3.1.2.1 Comunicación indirecta (Sistemas de pizarras y broadcasting).....	36
3.2 Métodos y protocolos para la coordinación	38
3.2.1 Red de contratos	41

3.2.2	Mecanismos de negociación	42
3.2.3	Mecanismos de votación	42
3.2.4	Mecanismos de subastas	43
3.2.5	Resolución de conflictos por arbitraje	44
4	Estructura de la propuesta de la arquitectura	45
4.1	Naturaleza del sistema	45
4.1.1	Descripción de los elementos robóticos	46
4.1.1.1	Pioneer P3-DX.....	46
4.1.1.2	Dron Parrot Bebop 2	47
4.2	Modelo de comunicación	48
4.2.1	Especificaciones físicas	48
4.2.2	Intercambio de mensajes	49
4.2.3	Organización de la comunicación.....	50
4.2.3.1	Sistema de pizarra empleado.....	51
4.3	Modelo de coordinación.....	53
4.3.1	Comportamiento y naturaleza de los agentes robóticos.....	53
4.3.2	Especialización y coordinación de acciones por planificación.....	54
4.3.3	Política cooperativa para resolución de conflictos.....	55
4.3.3.1	Reglas para la política cooperativa	55
4.4	Conclusiones de la arquitectura	58
4.5	Instrumentación de la arquitectura.....	59
4.5.1	ROS.....	59
4.5.2	Gazebo	61
5	Mapeo cooperativo de espacios.....	63
5.1	Fuentes de información en el mapeo.....	63
5.2	Representación de los mapas.....	64
5.3	Navegación y mapeo a través de sensores de rango.....	65
5.4	Propuesta de mapeo cooperativo de espacios.....	66
5.4.1	Pruebas en entorno simulado	67
5.4.2	Pruebas en entorno físico real.....	69
5.5	Cambios en la arquitectura de coordinación.....	70
6	Conclusiones y trabajo futuro.....	72
6.1	Trabajo futuro	74

Apéndice A Tipos de mensaje en el protocolo de comunicación	75
Bibliografía	78

ÍNDICE DE FIGURAS

FIGURA 1.1. ORGANIZACIÓN DE LA ROBÓTICA COLECTIVA	3
FIGURA 1.2. EJEMPLO DE TRABAJO DE MANTENIMIENTO EN UN TANQUE	5
FIGURA 1.3. CONFIGURACIÓN DE UN MANIPULADOR SERIAL-PARALELO	6
FIGURA 1.4. ARQUITECTURA DE LAS CÉLULAS EN EL TRABAJO ORIGINAL	6
FIGURA 1.5. MODELO DE COMUNICACIÓN ENTRE AGENTES DE VIGILANCIA	7
FIGURA 1.6. COMUNICACIÓN ENTRE ROBOTORS	8
FIGURA 1.7. ESTRUCTURAS DE COMUNICACIÓN TÍPICAS	9
FIGURA 2.1. EXPLORACIÓN Y AUTO LOCALIZACIÓN MULTI AGENTE	14
FIGURA 2.2. DESCRIPCIÓN GRÁFICA DE UN AGENTE	15
FIGURA 2.3. ORGANIZACIÓN MÚLTIPLE NATURAL (TOMADA DE [17])	17
FIGURA 2.4. RELACIÓN SIMBIÓTICA DE TIPO MUTUALISMO	18
FIGURA 2.5. GRUPOS DE PERSONAS EN SIMULACIÓN	18
FIGURA 2.6. GRAFO DIRIGIDO QUE MODELA EL FLUJO DE INFORMACIÓN ENTRE UN GRUPO SOCIAL	20
FIGURA 2.7. AGENTE MODIFICANDO SU COMPORTAMIENTO POR SU ENTORNO	21
FIGURA 2.8. COMPORTAMIENTOS COLECTIVOS CON ROBOTS DE BAJO COSTO	22
FIGURA 2.9. ESTRUCTURA GRÁFICA DE UNA RED NEURONAL ARTIFICIAL	23
FIGURA 2.10. PROBLEMA DE PATH PLANNING SIMULTÁNEO	25
FIGURA 3.1. CLASIFICACIÓN DE LOS SISTEMAS MULTI ROBOT [4]	29
FIGURA 3.2. MODELO DE INTERACCIÓN ENTRE MÚLTIPLES ROBOTS	31
FIGURA 3.3. EJEMPLO DE RANGO DE COMUNICACIÓN LIMITADO	32
FIGURA 3.4. PROCESO DE COMUNICACIÓN DIRECTA EMISOR-RECEPTOR	34
FIGURA 3.5. GRAFO DE COMUNICACIÓN NO DIRIGIDO COMPLETO (PRIMER CASO)	35
FIGURA 3.6. GRAFO DE COMUNICACIÓN NO DIRIGIDO FUERTEMENTE CONEXO (SEGUNDO CASO)	35
FIGURA 3.7. GRAFO DE COMUNICACIÓN NO DIRIGIDO FUERTEMENTE CONEXO CON UN CENTRO DE COMUNICACIÓN (TERCER CASO)	36
FIGURA 3.8. RED DE BROADCAST	37
FIGURA 3.9. FUNCIONAMIENTO DE UN SISTEMA DE PIZARRA	37
FIGURA 4.1. ROBOT MÓVIL P3-DX DE ADEPT MOBILE ROBOTS	46
FIGURA 4.2. CONFIGURACIÓN TÍPICA DE UN ROBOT MÓVIL DIFERENCIAL	46
FIGURA 4.3. MOVIMIENTO DE LAS LLANTAS EN UN MODELO DIFERENCIAL	47
FIGURA 4.4. ROBOT AÉREO NO TRIPULADO PARROT BEBOP 2	47
FIGURA 4.5. MODELO DE LAN PARA LA ARQUITECTURA	48
FIGURA 4.6. REPRESENTACIÓN DE LA PIZARRA POR TÓPICOS	51
FIGURA 4.7. PROCESO DE LA ESTRATEGIA DE COORDINACIÓN PIZARRA PARA EL PRIMER ROBOT	52
FIGURA 4.8. PROCESO DE LA ESTRATEGIA DE COORDINACIÓN PIZARRA PARA EL SEGUNDO ROBOT	52
FIGURA 4.9. PROCESO DE LA ESTRATEGIA DE COORDINACIÓN PIZARRA PARA EL TERCER ROBOT	52
FIGURA 4.10. SIMULACIÓN DEMOSTRANDO EL RANGO LÁSER DE LOS AGENTES TERRESTRES EN GAZEBO	54
FIGURA 4.11. CONFIGURACIÓN CORRECTA PARA LAS REGIONES SEGURAS DE LOS ROBOTS.	57
FIGURA 4.12. CONFIGURACIÓN INCORRECTA CON REGIONES SEGURAS TRASLAPADAS EN LOS ROBOTS	57
FIGURA 4.13. BLOQUEO DURANTE EL PROCESO DE EXPLORACIÓN DEBIDO A REGIONES SEGURAS	58
FIGURA 4.14. ESQUEMAS DE COMUNICACIÓN ENTRE NODOS DE ROS	60
FIGURA 4.15. SIMULACIÓN EN GAZEBO DE UN PIONEER P3-DX DE ADEPTMOBILE ROBOTS	61
FIGURA 4.16. EDITOR DE MUNDOS Y MODELOS EN GAZEBO	62
FIGURA 5.1. ESTIMACIÓN DE UNA POSICIÓN DE ACUERDO A UNA VELOCIDAD LINEAL Y UN ÁNGULO	63
FIGURA 5.2. REPRESENTACIÓN MÉTRICA DE UN MAPA A TRAVÉS DE SENSORES	64
FIGURA 5.3. REPRESENTACIÓN TOPOLÓGICA DE UN MAPA CON 7 LUGARES	64

FIGURA 5.4. REPRESENTACIÓN POR REJILLAS DE UN MAPA	65
FIGURA 5.5. ESPECIFICACIONES TÉCNICAS DE UN LÁSER HOKUYO	65
FIGURA 5.6. MODELO DE SENSADO DE ALTURA POR UN DRON	66
FIGURA 5.7. ENTORNO DE PRUEBAS DEL AMBIENTE 1	67
FIGURA 5.8. ENTORNO DE PRUEBAS DEL AMBIENTE 2	68
FIGURA 5.9. GRÁFICO COMPARATIVO DE LA PRECISIÓN EN DIFERENTES PRUEBAS	69
FIGURA 5.10. MUESTRA DE MAPEO EN ENTORNO SIMULADO (AMBIENTE 1)	69
FIGURA 5.11. MUESTRA DE MAPEO EN ENTORNO REAL	70

ÍNDICE DE TABLAS

TABLA 3.1. CATEGORÍAS DE ACTOS DE HABLA	30
TABLA 3.2. EJEMPLO DE MENSAJE CON UNA POSICIÓN ESPACIAL	31
TABLA 3.3. MÉTODOS DE COOPERACIÓN SEGÚN EL TIPO DE SISTEMA	40
TABLA 4.1. LISTA DE MENSAJES EN EL PROTOCOLO	50
TABLA 5.1. TABLA COMPARATIVA ENTRE PROMEDIOS DE PRECISIÓN PARA EL AMBIENTE 1.....	68
TABLA 5.2. TABLA COMPARATIVA ENTRE PROMEDIOS DE PRECISIÓN PARA EL AMBIENTE 2.....	68
TABLA 5.3. TABLA COMPARATIVA ENTRE PROMEDIOS DE PRECISIÓN PARA EL AMBIENTE 1.....	70
TABLA 5.4. TABLA COMPARATIVA ENTRE PROMEDIOS DE PRECISIÓN PARA EL AMBIENTE 2.....	70
TABLA A.1. ESTRUCTURA GENÉRICA DE UN MENSAJE	75
TABLA A.2. CAMPOS PARTICULARES PARA MENSAJE SALUDO	75
TABLA A.3. CAMPOS PARTICULARES PARA MENSAJE KEEP-ALIVE.....	75
TABLA A.4. CAMPOS PARTICULARES PARA MENSAJE DESCONEXIÓN.....	75
TABLA A.5. CAMPOS PARTICULARES PARA MENSAJE ASIGNACIÓN DE TAREAS.....	76
TABLA A.6. CAMPOS PARTICULARES PARA MENSAJE FINALIZACIÓN DE TAREAS	76
TABLA A.7. CAMPOS PARTICULARES PARA MENSAJE COMANDO DE MOVIMIENTO.....	76
TABLA A.8. CAMPOS PARTICULARES PARA MENSAJE MENSAJE DE CONFIRMACIÓN.....	76
TABLA A.9. CAMPOS PARTICULARES PARA MENSAJE POSICIÓN ESPACIAL	76
TABLA A.10. CAMPOS PARTICULARES PARA MENSAJE INFORMACIÓN SENSORIAL	77
TABLA A.11. CAMPOS PARTICULARES PARA MENSAJE ESTADO OPERATIVO	77
TABLA A.12. CAMPOS PARTICULARES PARA MENSAJE PETICIÓN DE INFORMACIÓN REMOTA	77
TABLA A.13. CAMPOS PARTICULARES PARA MENSAJE PETICIÓN DE PROCEDIMIENTO REMOTO	77

1 INTRODUCCIÓN

La robótica como una disciplina independiente comenzó en los años 1960. Los robots de aquella época operaban en espacios controlados y su primer dominio de aplicación fue la manufactura automovilística.

Con los avances en el campo de conocimiento de hardware y software, en la robótica moderna (a partir de los años 1980) nos encontramos con robots cuyas capacidades les permiten desplazarse en su entorno, así también realizar acciones para interactuar con este último, a diferencia de sus antecesores que únicamente trabajaban en un lugar confinado.

Las acciones a realizar son diversas, algunas clasificaciones de robots según el objetivo de los mismos [1] son:

- **Robots industriales.** Son robots usados en un entorno de manufactura industrial. Generalmente son brazos articulados con bases fijas, ya sea completamente autónomos o dependientes de un operador.
- **Robots de Servicio.** Son robots de uso doméstico que ayudan a las tareas básicas de los humanos. Pueden entrar muchos tipos de robots aquí, como las aspiradoras, limpiadores de piscinas, robots de vigilancia, entre otros más.
- **Robots para la medicina.** Son robots que asisten a las intervenciones quirúrgicas o a la movilización de pacientes en un hospital o clínica.
- **Robots militares.** Generalmente se incluyen robots preparados para el combate, ya sea para rescate y exploración, o vigilancia
- **Robots de hobby y competencia.** Son los robots usados en entretenimiento. Algunos ejemplos son los drones domésticos.

Independientemente de la utilidad que tuviera el agente robótico, siempre existió un robot que era el centro del sistema, por lo que éste ejecutaba todas las acciones solo. Incluso, en un inicio, sin algún tipo de percepción de su medio, es decir, sin sensores que le permitieran conocer su entorno.

En el año de 1985, R. Brooks propuso una elegante solución para robots móviles [2] que describe una arquitectura de estructura de control jerárquica donde una combinación de un sensor-actuador podía disparar las acciones de otro par similar en un nivel más bajo.

En otras palabras, combinando varios pares sensor-actuador, logró realizar un mecanismo que ejecutaba acciones de manera autónoma de acuerdo a las reacciones de cada par, tal como si fueran engranes.

Con el paso del tiempo, la tecnología en la robótica fue avanzando de manera considerable, y junto con una progresión en otras áreas como la inteligencia artificial distribuida, fue una curiosidad comenzar a trabajar con sistemas donde se colocaba a un equipo de robots para estudiar las capacidades colectivas, todo esto al inicio de los años 1990.

Un sistema de múltiples robots tiene muchas ventajas inherentes como sensado y acciones distribuidas, reconfigurabilidad basada en tareas, diseños más simples para cada individuo en un equipo y un diseño de sistema más confiable debido a la redundancia de agentes disponibles [3].

1.1 GRUPOS DE ROBOTS

Cuando un agente robótico convive con otros agentes en un mismo entorno, el concepto de acción adquiere un nuevo significado. El robot ya no se puede concebir como una entidad aislada y única dentro de un entorno que responde exclusivamente a sus estímulos. Ahora dicho entorno incluye explícitamente otros elementos interactivos, que por sí mismos, ejecutan acciones que responden igualmente al ambiente.

En un sistema de esta naturaleza, las acciones e influencias de los agentes deben tomarse en cuenta y ser consideradas de forma individual y de manera grupal. Se podría afirmar por lo tanto que, así como la acción es esencia de los agentes, la idea de interacción es la esencia de los sistemas con múltiples agentes.

En el momento en que un agente interactúa con otros agentes, esta interacción, que es la principal fortaleza y justificación de este tipo de sistemas, se vuelve también una fuente importante de problemas [4].

Las interacciones entre agentes pueden ser positivas, permitiéndoles cooperar y ejecutar tareas que van más allá de lo posible con sus capacidades individuales. Sin embargo, también pueden tener efectos negativos por la generación de interferencias y conflictos entre ellos. Los conflictos pueden estar motivados por sus distintos puntos de vista sobre el problema a resolver, por sus intereses personales o sencillamente porque deben compartir recursos o espacio escasos.

La posible existencia de conflictos requiere de mecanismos para resolverlos, o mejor aún, evitarlos, lo que implica en general la realización de una serie de actividades adicionales. Estas actividades adicionales se presentan en formas de control de grupos.

Existen dos tipos de enfoques para el control de grupos de robots móviles. El primero, el enfoque convencional, es dependiente de las habilidades de los robots, creando así modelos intrínsecos relacionados a la tarea en específico. La comunicación en este caso es explícita con otros robots u otros agentes (humanos incluidos). En el segundo enfoque, designado como el paradigma de enjambres (swarm paradigm), la interacción local entre los robots y con el entorno en el que se están desarrollando, gobiernan ambos la ejecución de las tareas administradas [5].

Las sociedades de insectos proveen la inspiración para crear equipos sinérgicos de robots en cuando al enfoque de enjambres. El enfoque *Animat*, una metodología innovadora, imita sistemas que provienen de comportamientos naturales en animales en general [6].

Las áreas de la robótica colectiva, en las cuales los investigadores se han enfocado, incluyen el ordenamiento, búsqueda de objetivos (foraging), movimientos de formación, transporte grupal y construcción colectiva.

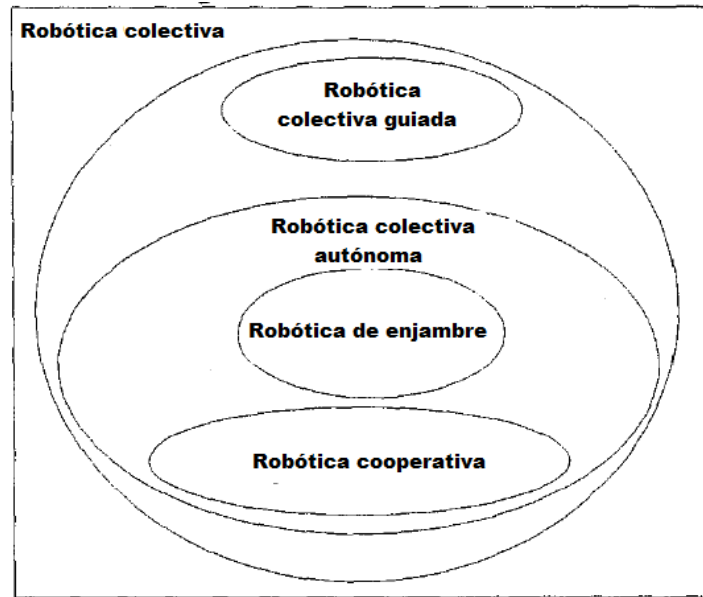


Figura 1.1. Organización de la robótica colectiva

En la Figura 1.1 se aprecian las sub áreas en las que se divide la robótica colectiva. En particular, la primera clasificación es la robótica guiada por humanos y la robótica autónoma. Dentro de esta última tenemos la robótica colectiva convencional que habíamos mencionado anteriormente, junto con la robótica de enjambres.

Es necesario tomar en cuenta el dominio en que un agente será aplicado, para efectuar un análisis correcto de los mecanismos de interacción entre agentes ya que, no serán iguales las interacciones entre agentes que cooperan conjuntamente en la realización de una tarea u objetivo común, que en aquellos dominios donde el objetivo de un agente es derrotar o destruir al otro.

También existirán casos intermedios donde si bien existe una idea de objetivo global o social, los agentes se comportarán en forma egoísta intentando maximizar el beneficio propio, sin importar el bienestar colectivo. En cada uno de estos casos, los mecanismos de interacción serán completamente distintos, ya que los supuestos en los que se basan difieren completamente.

1.2 PROBLEMAS DE CONTROL DE GRUPOS DE ROBOTS

Hay dos conceptos cruciales que deben ser tomados en cuenta en el diseño de un sistema de múltiples robots. Ellos están definidos por R. Brooks como [2]:

1. Posición: Los robots están situados en el mundo en posiciones absolutas. No tienen que lidiar con descripciones abstractas, más que con el “aquí” y “ahora” del entorno que directamente influyen el comportamiento del sistema.
2. Encarnación: Los robots tienen estructuras que podemos considerar “cuerpos” que interactúan con el mundo directamente. Sus acciones son parte de una dinámica con el mismo, y las acciones tienen retroalimentación inmediata en las propias “sensaciones” del robot.

Otros requerimientos que guían el diseño incluyen requerimientos de comunicación, organización social, tareas y características del entorno [7].

Si se observan los requerimientos que origina un grupo de robots, es fácil inferir que es necesario tener una jerarquía de control. Un primer nivel de control para el agente físico y sus acciones como individuo, y otro a nivel global para el cumplimiento de la meta u objetivo general [8].

La parte más interesante yace en el control individual. Si un individuo tiene un comportamiento ‘individualista’, lo que sucederá es que probablemente haya más tipos de interferencias entre los robots que si tuviera un comportamiento ‘cooperativo’.

En la práctica, este tipo de diferencias se resuelve de una forma sociológica, es decir, ya sea con sistemas de democracia, o con sistemas de sorteo [4], para poder realizar las tareas individuales con el menor número de interferencias y alcanzando el objetivo en un tiempo prudente o de manera eficiente.

1.3 ESTADO DEL ARTE

En general, el fuerte impacto entre los años 1980 y 1990 fue bastante fuerte en la robótica colectiva. Muchos trabajos son bastante interesantes para su revisión, sin embargo, en esta sección se citan trabajos que tienen contenido similar al tema propuesto aquí, o contienen elementos que son útiles para el desarrollo.

Los tópicos de interés particular en estos trabajos de robótica colectiva son:

- Cellular Robotic System (CEBOT), trabajo de Fukuda, *et al.* [9]. Es un sistema robótico capaz de reconfigurarse a sí mismo sobre la marcha.
- Sistema multi agente de vigilancia, trabajo de Vallejo, *et al.* [10]. Sistema compuesto de agentes que exploran un entorno desconocido con cámaras en redundancia y canales de comunicación jerárquicos.
- ACTRESS (ACTor-based Robots and Equipments Synthetic System), trabajo de Asama, *et al.* [11]. Un sistema multi robot en el que sus autores se enfocan en el diseño de una arquitectura orientada a la cooperación.

1.3.1 CEBOT

En este trabajo, los autores proponen un modelo de un sistema con un robot manipulador, con la característica especial que el manipulador está conformado en su estructura por

celdas robóticas que pueden reconfigurarse en tiempo real. El trabajo únicamente se postuló en forma de simulación, el robot real no se construyó al momento de generar la idea.

Los robots como CEBOT caen en una categoría de robots conocida como DRRS (Dynamically Reconfigurable Robotic System o Sistema Robótico Reconfigurable Dinámicamente). Este tipo de robots son famosos por lograr adaptarse a diferentes tareas cambiando su estructura física.

En el caso concreto de CEBOT, se considera un conjunto de células robóticas que se comunican entre ellas para adoptar ciertas formas de acuerdo al papel que tenga que cumplir el robot.

Uno de los aspectos importantes a destacar con este manipulador, es el hecho de poder desarmarse para ser transportado de manera más simple. Además de ello, también permite reconfigurar los grados de libertad con los que trabaja dicho manipulador.

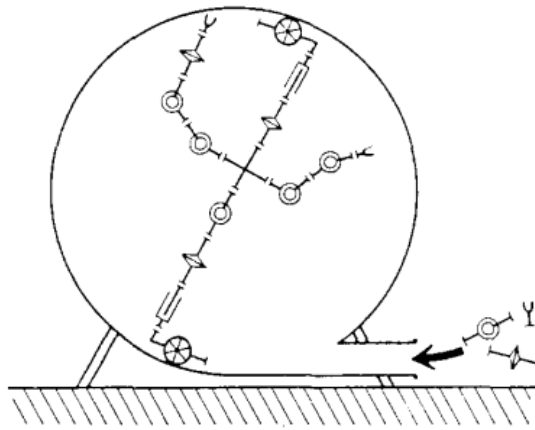


Figura 1.2. Ejemplo de trabajo de mantenimiento en un tanque

En la Figura 1.2 se ilustra un posible uso que se le da a un robot manipulador reconfigurable. Las células son transportadas a través de un pequeño tubo, para posteriormente autoconfigurarse dentro del tanque en un espacio más grande, con el fin de poder efectuar tareas de limpieza dentro de él. Esta tarea ilustra la practicidad de un sistema DRRS, y una enorme ventaja sobre sistemas robóticos fijos.

La arquitectura física del robot manipulador se conforma de tres elementos:

- **Nivel 1.** Células de unión (articulaciones para flexión, rotación y desplazamiento) y células móviles (llantas, tracción).
- **Nivel 2.** Células de ramaje in dos y tres dimensiones espaciales. Son células para la longitud del brazo entre articulaciones y la transformación de la orientación; y células de poder para efectuar tareas de fuerza.
- **Nivel 3.** Células de trabajo (efectores finales) y demás que son requeridas para tareas de propósito no típicas.

La combinación y desprendimiento entre células se lleva a cabo por las células móviles que pueden transportarse a sí mismas, o a las células de articulación que fueron combinadas anteriormente como un manipulador.

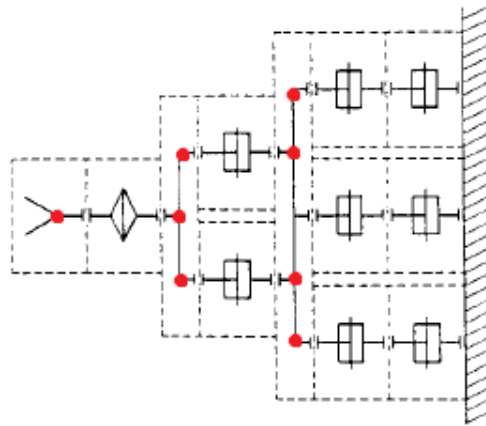


Figura 1.3. Configuración de un manipulador serial-paralelo

En la Figura 1.3 se presenta la configuración de un manipulador de tres divisiones, con un efector final en cada punta. Los puntos rojos representan células de nivel 1, las cajas representan células de nivel dos, y por último, las puntas del lado derecho representan células de nivel 3.

Si bien en este trabajo no se detallan aspectos sobre la comunicación de las células detalladamente, lo que los autores proponen son sistemas de selección de células candidatas, las cuales compiten para ver quién es la más apta de acuerdo a la posición relativa entre componentes que deben ensamblarse.

Es por esta razón que cada una de ellas debe poseer las siguientes características:

- Capacidad de posicionarse espacialmente (Odometría)
- Capacidad de comunicación
- Unidad central de procesamiento como individuo

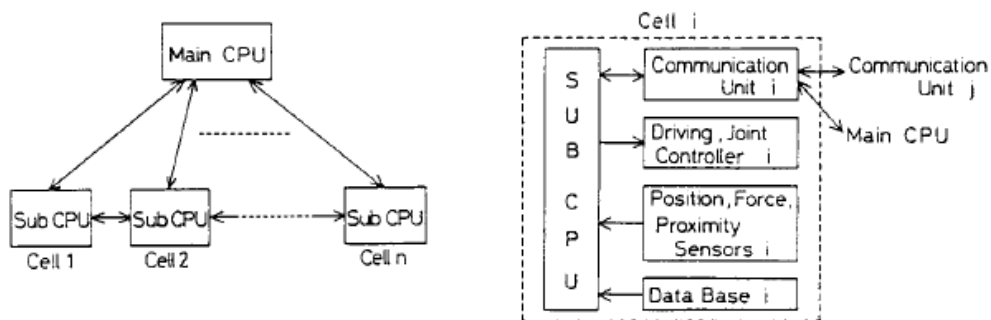


Figura 1.4. Arquitectura de las células en el trabajo original

El sistema en sí, por la descripción de los autores, propone un sistema centralizado que determina las órdenes a seguir por cada célula. En la Figura 1.4 extraída del trabajo original, se puede apreciar la arquitectura de las células, la cual requiere una unidad central de procesamiento principal para realizar las decisiones.

Aunque es un trabajo muy bueno, los autores no dan muchos detalles sobre su posible (en ese momento) implementación del robot de manera física. Los conceptos relevantes son que se comienza a realizar una arquitectura centralizada, aunque no sea descrita explícitamente como tal, pero es una de las primeras aproximaciones en el área de los sistemas multi robot.

1.3.2 Sistema multi agente de vigilancia

En este trabajo, los autores proponen una arquitectura multi agente que da soporte a un conjunto de robots para tareas de vigilancia, tales como la exploración de entornos. En el trabajo, ellos describen la arquitectura del sistema de manera detallada como un sistema híbrido (una parte centralizada, una parte distribuida), en la que se asignan las acciones y los medios de comunicación a través de roles que cumple cada agente.

La parte centralizada de este sistema la conforma la asignación de roles, que se realiza mediante las capacidades motrices de cada agente; mientras que también se recolectan datos del entorno a través de los sensores.

El modelo de comunicación utilizado por este tipo de robots es un esquema basado en canales activados por eventos que todos los agentes pueden consultar dependiendo del rol que tengan.

Para la asignación de roles, los agentes están clasificados en dos tipos principales:

- Los agentes superiores
- Los agentes pertenecientes a celdas de trabajo

Los agentes de celda poseen sus propios canales que funcionan para cada celda en particular y sólo los miembros de dicha celda pueden acceder a él. Para la comunicación entre celdas existe un canal para los agentes superiores, dominando cada uno una celda e informando cada dato recibido a través de dicho canal.

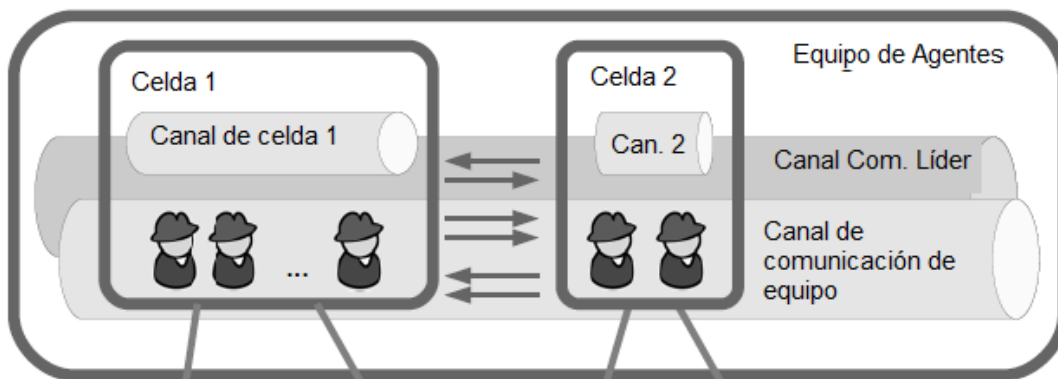


Figura 1.5. Modelo de comunicación entre agentes de vigilancia

En una variante implementada por ellos mismos intentaron realizar un segundo canal de comunicación de equipo pero con comunicación regulada para hacer llegar los mensajes globales más rápido, como los mensajes de metas cumplidas.

En este trabajo se resalta la importancia de definir dentro de la arquitectura un esquema de mensajes que permite a los agentes saber cuándo acontece algo relevante dentro del entorno al momento de que uno de ellos se percata del suceso.

1.3.3 ACTRESS

ACTRESS (ACTor-based Robots and Equipments Synthetic System) es un sistema robótico autónomo y distribuido compuesto de elementos multi robóticos. Cada elemento que proveen los autores se entrega con funciones para hacer decisiones con el entendimiento del objetivo de las tareas (funciones objetivo y de optimización), reconocimiento del entorno que lo rodea (conjuntos de sensores), acciones (a través de actuadores y su propio movimiento), y capacidad de manejar sus propias condiciones, y de comunicarse con otros componentes.

Para administrar múltiples elementos, los autores emplearon protocolos de comunicación, y específicamente una arquitectura híbrida, de tal manera que los robots pueden ejecutar tareas de manera cooperativa, tomando en cuenta que cada uno de ellos también puede tomar decisiones de manera individual.

Aquí también, los autores decidieron crear un tipo de robot especial para el sistema, llamado “micromouse” o microratón. Por lo tanto este tipo de sistema es homogéneo en cuanto a los elementos robóticos que lo conforman.

Los dos elementos principales de este sistema, por lo cual recibe su nombre, son:

- Actores
- Mensajes

Esto es una manera de formalizar los términos dentro del sistema, realmente todo confiere a agentes y la manera de comunicarse entre ellos. Los actores son nombrados “Robotor”.

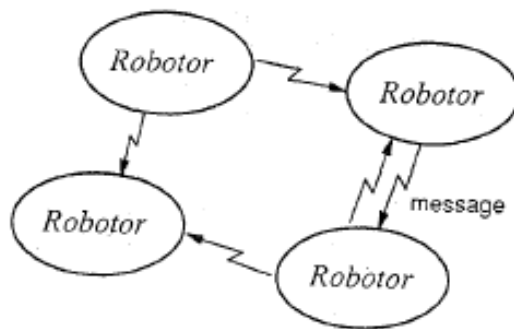


Figura 1.6. Comunicación entre Robotors

Con respecto a los *Robotors*, cada *robotor* es un componente autónomo, el cual se asume que debe tener al menos dos funciones básicas.

1. La habilidad para tomar decisiones, entender el objetivo de las tareas, reconocer los entornos alrededor de él, actuar y administrar sus propias condiciones.
2. La posibilidad de comunicarse con otros componentes para realizar tareas paralelas, incluyendo evitar las interferencias en los movimientos de los componentes del sistema, o para tareas cooperativas donde sea necesario el paso de información.

Lo importante de ACTRESS, es que define completamente la arquitectura del sistema, el tipo de mensajes que van a mediar entre los robots, y la manera física de obtenerlos.

Para la comunicación entre los *Robotors*, los autores aceptan el protocolo de comunicaciones de redes estándar, estructurando una red LAN, considerando que existen computadoras personales trabajando dentro de ella, utilizando el modelo OSI (Open Systems Interconnection). En la Figura 1.7 se aprecia los tres tipos de estructuras de comunicación típicas dentro de los modelos LAN con respecto al sistema.

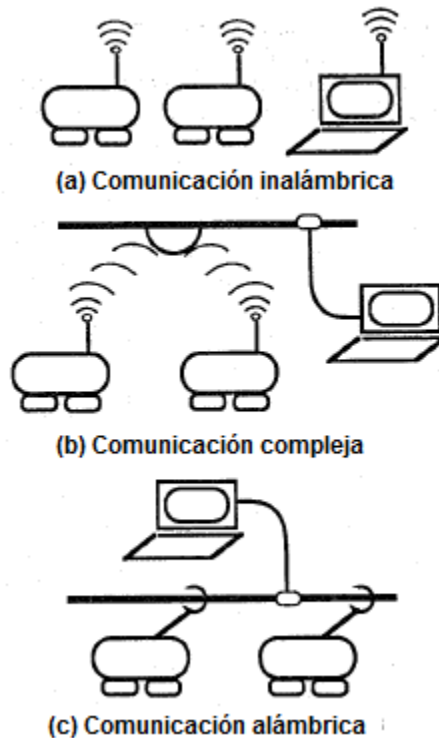


Figura 1.7. Estructuras de comunicación típicas

Lo importante de este trabajo es que los autores especifican explícitamente el protocolo de mensajes que diseñaron para su sistema. La arquitectura de la información corresponde muy bien a uno de los lenguajes de robot conocidos, que fue expresado como una jerarquía de niveles de comando, objetos y tareas [12].

Los mensajes pueden transmitir diferentes cosas, por lo que ellos decidieron crear una capa de 5 niveles de mensajes, los cuales se describen a continuación.

1. **Nivel de Control.** Son mensajes dirigidos a los componentes físicos de los elementos, para activar algún efector final, o recibir información técnica de un sensor.
2. **Nivel Físico.** Contienen mensajes con datos físicos de los robots, como la posición, velocidad y fuerza. Se envían parámetros de dimensión.
3. **Nivel Procedural.** Son mensajes que permiten invocar procedimientos en los robots.
4. **Nivel de Conocimiento.** Son mensajes específicos de ACTRESS que permiten mandar información que no recae en los otros tres niveles, para interpretación exclusiva de las computadoras personales a cargo del sistema.
5. **Nivel Conceptual.** Se transmiten las intenciones y los objetivos en este tipo de mensajes. En este tipo también se transmiten los mensajes cuando se cumplen los objetivos programados.

En general, es difícil clasificar todos los tipos de mensajes posibles en un sistema multi agente en ciertos tipos, pero es necesario debido a que puede dar una idea de lo que se espera recibir en él, y la manera de cómo procesarlos.

En especial, ACTRESS tiene una ventaja enorme sobre las demás propuestas por esta arquitectura tan detallada que los autores proponen; ya que, como se verá más adelante, es necesario definir explícitamente la información que los agentes van a intercambiar entre ellos.

1.4 JUSTIFICACIÓN Y CONTRIBUCIONES DEL TRABAJO

Cuando se propuso la idea de una arquitectura para la coordinación de múltiples robots, la idea que se tenía en mente era que el diseño de ésta fuera completamente independiente de los elementos que lo conformen, es decir, un sistema conformado por elementos heterogéneos.

Incluso, el punto de partida de la investigación fue el trabajo realizado dentro del laboratorio MOVIS en la Facultad de Ciencias de la Computación de la Benemérita Universidad Autónoma de Puebla [4], el cual describe diversas técnicas para coordinar robots y sus principales ventajas. Esta vez se pretende con todo este conocimiento, diseñar y probar los mejores casos para un sistema multi robot de naturaleza heterogénea y explorar sus comportamientos.

En los trabajos descritos en la sección anterior, se puede apreciar que una arquitectura siempre es diseñada sobre los elementos robóticos que se van a manejar. Aunque esto sea casi siempre un requisito, en el presente trabajo se tiene la idea que existen partes de una arquitectura que pueden ser reutilizadas en otros sistemas, según los robots que se utilicen.

Como se observó, el primer paso en un diseño de sistemas multi robot era decidir el tipo de robots que se van a utilizar y posteriormente construir una arquitectura completamente *ad-hoc* al sistema. Esto nos hace dependientes de la implementación física.

Por supuesto que lo anterior no representa un problema muy grande, salvo en los casos en los que sea necesario agregar nuevos tipos de elemento al sistema, los cuales no estaban contemplados en un inicio.

Es por esto que en el presente trabajo se propone primero el diseño de una arquitectura, para posteriormente añadir los elementos robóticos necesarios, clasificándolos en tres tipos de robots de acuerdo al medio por el cual se desplazan.

- Robots terrestres móviles
- Robots aéreos no tripulados
- Robots acuáticos

Uno de los puntos importantes a destacar en el diseño de una arquitectura de coordinación también son las tareas. Muchas veces las arquitecturas se diseñan en torno a ellas, y en caso de existir intercambio de mensajes, son muy específicos al tipo de tarea que se esté realizando.

En el caso de la arquitectura propuesta, se toman en cuenta mensajes genéricos, por lo que las tareas se diseñan encima de él. Esto pone un ligero predicamento en las tareas a realizar, ya que si se diseñan las tareas por encima de arquitecturas genéricas, no pueden ser tareas demasiado específicas y atadas a algún tipo de robot en especial, aunque esto sucede muy poco generalmente.

El objetivo general del trabajo es

1.5 ORGANIZACIÓN DEL ESCRITO DE TESIS

Para dar un breve resumen acerca de lo que se abordará a lo largo del presente trabajo se presentan a continuación un listado de cada capítulo con una breve reseña que lo describe.

- **Capítulo 1: Introducción:** En este capítulo ya visto se tratan aspectos generales del trabajo de tesis, algunos alcances, la justificación y la organización del mismo. Se presentan términos que serán definidos con más detalle a lo largo del desarrollo de los capítulos 2 y 3, los cuales pueden considerarse el Marco Teórico del trabajo.
- **Capítulo 2: Sistemas multi robot y sistemas multi agente:** Se tratan temas acerca de la teoría de sistemas multi robot y sistemas multi agente en general. Se presentan al lector las propuestas de organización para cada agente, los conceptos tomados de la organización social, y los elementos clave que forman un sistema de esta naturaleza.
- **Capítulo 3: Arquitecturas para la coordinación multi robot:** Se presentan los elementos que debe describir una arquitectura para coordinar varios agentes en un mismo espacio, así también la naturaleza de las tareas para poder ser aplicables en sistemas multi robot, además de ejemplificar y comparar las diversas aproximaciones que se exponen en teoría.
- **Capítulo 4: Estructura de la propuesta de la arquitectura:** Se expone el diseño a fondo, exponiendo detalladamente la estructura de los elementos de coordinación y de comunicación entre los elementos presentes en la arquitectura. Se agregan definiciones específicas que no se han detallado en los capítulos anteriores.
- **Capítulo 5: Mapeo cooperativo de espacios:** Se introduce la tarea a resolver, además de los conceptos subyacentes para el entendimiento y resolución de la misma. Se presentan los resultados obtenidos a partir de la aplicación de la arquitectura sobre

simulaciones y pruebas con robots reales en el mapeo de espacios. En tareas con pocos elementos se presenta la comparativa de la misma implementación en ambos medios (simulado y físico).

- **Capítulo 6: Conclusiones y trabajo futuro:** En general, se destacan los puntos importantes observados a través de la investigación. Posteriormente se proponen algunos ejes de mejora, para darle continuidad dentro y fuera del entorno de desarrollo del presente trabajo. Se destaca la ventaja que tuvo el encontrar un framework de desarrollo para la comunidad del laboratorio MOVIS.

2 SISTEMAS MULTI ROBOT Y SISTEMAS MULTI AGENTE

El siglo XX fue testigo del nacimiento del término robot, que se define en el Diccionario de la Real Academia de la Lengua Española como la máquina o ingenio electrónico programable capaz de manipular objetos y realizar operaciones antes realizadas sólo por personas. Así mismo, se incluye una definición para Robótica como la disciplina que aplica la computación al diseño y empleo de aparatos que en sustitución de personas, realizan operaciones o trabajos, por lo general en instalaciones industriales.

Como ya se había introducido en el capítulo anterior, los sistemas robóticos desde su implementación, e incluso hasta el momento, generalmente cuentan con un único agente robótico, dotado de capacidades para procesar su entorno y manipularlo. Sin embargo, las capacidades de un sólo robot pueden verse limitadas debido a que físicamente sólo puede ocupar un lugar en el espacio.

Imaginemos un escenario donde una persona deba acarrear cajas desde un camión de descarga a una bodega localizada a 200 metros, distancia que por alguna razón no puede ser acortada. Para esto, sólo se puede acarrear una caja a la vez, por lo que se debe regresar por otra y volver a hacer otro viaje.

El modelo parece simple y fácil de implementar, sin embargo, si queremos ahorrar tiempo entonces podemos considerar la idea de que emplear a más de una persona puede ser conveniente en la reducción de tiempos.

Precisamente esta es la idea que inspira a los investigadores desde los años 1980 a emplear más de un agente en un sistema robótico, con el primer objetivo de reducir tiempos de ejecución en las tareas que un único robot debía realizar.

2.1 DE UN SOLO ROBOT AL CONJUNTO DE ROBOTS

Una vez que los investigadores plantean la idea de colocar varios agentes robóticos en el mismo plano espacial, se observa y se infiere que un sistema de esta índole puede ganar diversas ventajas, además del tiempo de ejecución [13] [14].

En [15], los autores demostraron que, en un mismo espacio, un conjunto de robots puede localizarse a ellos mismos más rápido y de manera más certera si intercambian información sobre sus posiciones cuando, por medio de sus sensores, detectan la presencia de alguno de ellos. En la Figura 2.1 se aprecia la exploración con la auto localización de varios agentes robóticos.

En el párrafo anterior se introduce un concepto clave en un sistema multi robot: la redundancia. La redundancia consiste en tener varios recursos similares que pueden ayudar a tener disponible información que, puede corroborarse como verdadera más rápido si se encuentra de manera repetida en más de un agente.

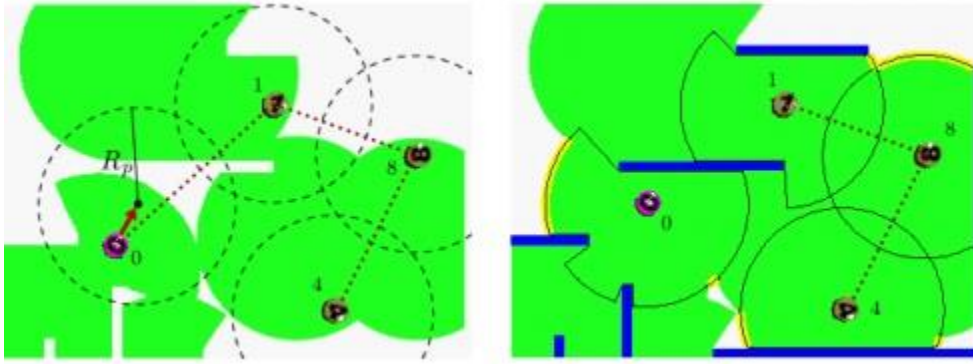


Figura 2.1. Exploración y auto localización multi agente

Es claro que un sistema de un solo robot no puede tener redundancia a nivel de agentes, aunque puede tener redundancia en los sensores y actuadores que posea, aunque es claro que las ventajas de un enfoque a otro son distintas. En el enfoque de redundancia por múltiples agentes, podemos observar un mismo suceso de diferentes puntos y ángulos, lo cual nos puede dar información adicional del suceso. En el enfoque de redundancia por múltiples sensores y actuadores, lo que tenemos es información traslapada desde un mismo punto en uno o más ángulos, lo cual limita la visión del mismo suceso.

Otro punto importante en la redundancia por elementos múltiples es que podemos utilizar piezas con menor costo, a comparación de utilizar un único agente de mayor costo, mientras se adquiere una tolerancia a fallos más grande [13].

Ahora bien, los sistemas multi robot pueden verse como un subconjunto de los sistemas multi agente, ya que la única diferencia es la naturaleza de los mismos agentes la que cambia, respetando las demás normas en cuanto a la organización y comunicación de agentes.

2.2 AGENTES

Tanto se ha hecho mención en el presente trabajo sobre la palabra 'Agente', mientras que hasta el momento no se ha palpado alguna definición concreta acerca de ello. Si bien no hay una definición clara en el conjunto de literatura de sistemas multi agente, por lo que hemos tratado sobre la asociación agente-robot podemos concluir algunas cosas que nos ayudarán a construir una definición de un agente.

El término robot refiere a un conjunto de piezas mecánicas ordenadas de cierta forma que permite una coordinación de fuerzas, que desemboca en una acción provocada por movimiento de piezas llamadas actuadores. A su vez también existen robots que pueden percibir información del entorno en el que se encuentran.

Esencialmente, mientras hay un consenso general que la *autonomía* es el centro de la noción de un agente, existe poca aceptación detrás de esto. Parte de esta dificultad es que varios atributos asociados con los agentes son de diversa importancia para los diferentes dominios en los que se aplican. Es por ello que, para algunas aplicaciones, la habilidad de los agentes

para *aprender* de sus experiencias es de suma importancia; para otras aplicaciones, esta habilidad no solo es de poco interés, incluso llega a ser indeseable [16].

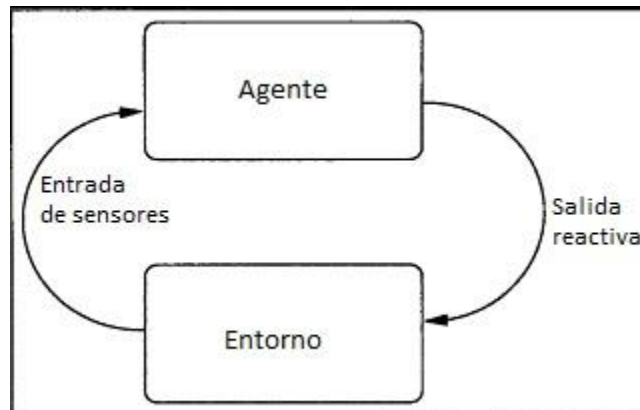


Figura 2.2. Descripción gráfica de un agente

En términos generales, un agente es un sistema o pieza de programa computacional que está situada en algún *entorno*, y que es capaz de realizar *acciones autónomas* en dicho entorno para cumplir sus objetivos por los cuales fue diseñado.

Es una definición bastante abstracta adaptada a todos los campos en los que un agente puede encajar. Anteriormente se había utilizado la palabra 'entorno' sin haberla definido específicamente. En el caso del ámbito robótico, el entorno se refiere al espacio físico dimensional en el que el agente va a poder ejecutar desplazamientos.

Las 'acciones', palabra que también se había mencionado con anterioridad, se refieren en este caso particular al movimiento que va a ejecutar el agente, ya sea sobre sí mismo o sobre otros elementos dentro del entorno. Se habla de una 'acción autónoma' cuando el movimiento que ejecuta el agente no necesita algún tipo de intervención humana o de otro sistema. Es decir que tienen control sobre su estado interno y sobre su comportamiento.

En la Figura 2.2 se da una vista general de la estructura de un agente con su entorno. En este diagrama, se puede observar que las salidas de las acciones que ejecuta un agente (Salida reactiva) afectan su entorno. En la mayoría de los dominios de complejidad razonable, un agente no va a tener el control completo de su entorno. A lo sumo tendrá un control parcial del mismo, lo suficiente para influenciarlo. Del punto de vista de un agente, lo anterior significa que la misma acción realizada dos veces en aparentemente las mismas circunstancias parece tener efectos completamente diferentes, y en particular, puede fracasar en tener el efecto deseado.

La posibilidad del fallo en los efectos de un agente indica entonces que, para todos los agentes, incluso en el más trivial de los entornos, deben estar preparados para *fracasar*. Un entorno en el que los agentes tengan posibilidad de fallar, se le denomina *no determinístico*.

Normalmente, un agente tiene un repertorio de acciones disponible para él. Este conjunto de posibles acciones representa la *capacidad efectora* del agente. Hay que recordar que no todas las acciones pueden estar disponibles en todas las situaciones. Por ejemplo, una acción

“levantar caja” sólo es aplicable en los casos donde exista una caja cuyo peso sea lo suficientemente pequeño para que el agente la pueda levantar. Por lo tanto, las acciones tienen precondiciones asociadas con ellas, las cuales definen las posibles situaciones en las que pueden ser aplicadas.

2.3 SISTEMAS MULTI AGENTE

Ahora que sabemos qué es un agente como unidad, ¿Qué sucede cuando mezclamos en un mismo entorno varios agentes?

Los Sistemas multi agente son una rama de la Inteligencia Artificial Distribuida (Distributed Artificial Intelligence, DAI), que a su vez es un área de la Inteligencia Artificial.

La DAI, que puede definirse como el estudio de la concurrencia en Inteligencia Artificial, se divide tradicionalmente en dos campos: la resolución de problemas distribuidos (Distributed Problem Solving, DSP) y los sistemas multi agente (Multi-agent Systems, MAS). Ambas se describen más detalladamente a continuación:

- La investigación en DSP se centra en establecer la solución de un problema usando varios agentes; en este esquema se descompone el problema inicial en tareas y se asignan a los distintos agentes. En DSP se supone que cada agente desea colaborar para obtener el resultado final, es decir, están predispuestos a la cooperación.
- División del trabajo en porciones, que se asignen a los elementos del sistema, de manera que se concluya antes y, por lo tanto, aumente el rendimiento global.
- Los sistemas multi agente también intentan resolver un problema mediante varios agentes. Sin embargo, frente a los DSP, los elementos de un sistema multi agente no están inicialmente hechos para cooperar, **sino que poseen sus propios intereses y objetivos.**

Como vemos, los agentes tienen comportamientos individuales que, cuando son puestos en marcha en el mismo entorno, parece ser que pueden imitar sociedades y colonias. Por ejemplo, las colonias de hormigas tienen obreras, soldados y una reina; cientos de hormigas pueden cargar un gusano de tierra muerto de un lugar a otro.

Como podemos observar con el ejemplo de las hormigas, los agentes no sólo pueden realizar acciones sociológicas, sino también pueden adquirir roles junto con tareas y acciones específicas a dicho rol.

En cuanto a la naturaleza de los agentes, podemos clasificar los sistemas multi agente en tres diferentes taxonomías [4]:

- Apatía o indiferencia. Donde, la tarea de cada agente es independiente de la de los demás, y puede ser llevada a cabo sin relacionarse con el resto de integrantes, y sin que una posible colaboración aporte beneficios.
- Cooperación. Es la asociación de agentes para lograr un fin común.

- Competencia. Los objetivos de los agentes del sistema son incompatibles entre sí. Aquí entrarían las técnicas de persecución-evasión, o juegos competitivos como el fútbol robótico.

Generalmente un sistema gana algún tipo de taxonomía, que incluso puede ser híbrida, de acuerdo a los objetivos que se deban cumplir. Una vez establecida, la taxonomía obliga a tomar ciertos tipos de organización entre los agentes para poder llevarla a cabo.

2.4 CONTROL COOPERATIVO

Cuando se introducen objetivos y taxonomías en un sistema, es necesario tener mecanismos de mediación entre los agentes, que permitan llevar, lo más eficiente posible, las actividades necesarias para cumplir las metas.

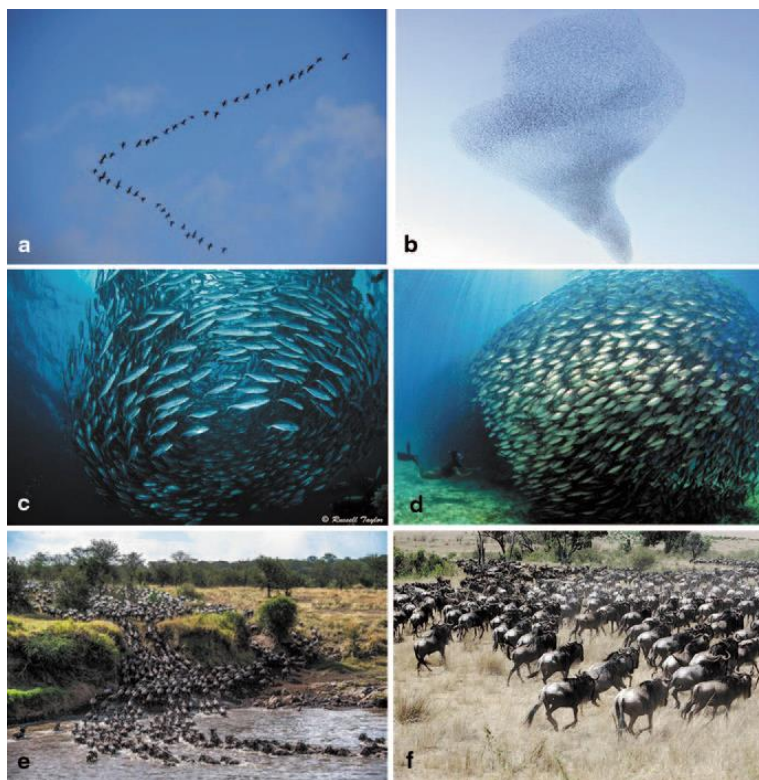


Figura 2.3. Organización múltiple natural (Tomada de [17])

En la naturaleza se han observado movimientos colectivos de grupos sociales de animales, los cuales son vistas interesantes. Cada individuo tiene sus propias inclinaciones y movimientos; aun así, el movimiento en conjunto hace parecer que el grupo es una única entidad con sus propias leyes de movimiento, psicología y respuestas a eventos externos. Flotas de aves, escuelas de peces, y hordas de animales son entidades en conjunto que se unen para desplazarse, como se ilustra en la Figura 2.3.

Cabe aclarar que este tipo de control natural colectivo es en la misma especie, puesto que existen otros tipos de cooperación y coordinación conocidas. La simbiosis es un claro

ejemplo de ello. Definida como la interacción biológica cercana y a largo plazo entre dos especies diferentes de animales.

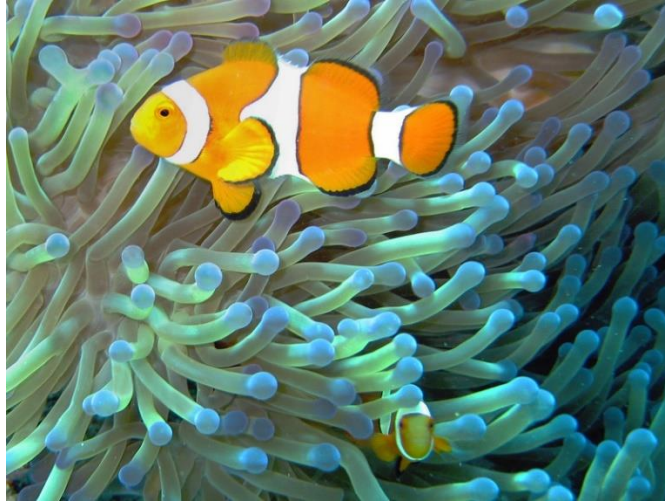


Figura 2.4. Relación simbiótica de tipo mutualismo

En la simbiosis, existen tres tipos de relaciones biológicas entre especies:

- Mutualismo. La interacción de ambas especies termina por beneficiar a todos los participantes de dicha interacción.
- Comensalismo. Una de las especies interactuando resulta beneficiada mientras la otra resulta indiferente de la relación.
- Parasitismo. Una de las especies resulta beneficiada, perjudicando en cierto grado a la otra especie.

En términos de agentes, podemos utilizar los conceptos de hordas para referirnos a conjuntos de agentes con las mismas características, los cuales adquieren habilidades propias para coexistir y trabajar cooperativamente. Véase la Figura 2.4.



Figura 2.5. Grupos de personas en simulación

Por parte de la simbiosis, es posible colocar agentes de diferentes naturalezas a cooperar entre ellos, para obtener un beneficio mutuo o individual de alguno de ellos.

A lo largo de varias investigaciones recopiladas en [18] se ha llegado a la conclusión de que existen diversas técnicas en las que podemos realizar control de manera cooperativa en un sistema multi agente:

- Centralizado
- Distribuido
- Mixto

En el caso de los ejemplos mostrados en la Figura 2.1, existen tipos de relaciones centralizadas, como en el caso de las parvadas que son guiadas por una cabeza de la formación. El ave al frente lleva la dirección del conjunto. En caso de ocurrirle algo a la cabeza, otra ave en jerarquía preestablecida puede tomar el control de la formación.

Para los bancos de peces y otros tipos de formaciones, generalmente se tiene un control mixto, en el que existen partes que organizan la formación, pero también los seres vivos son capaces de tomar decisiones de acuerdo a los obstáculos presentes y otro tipo de interferencias.

Con respecto a las relaciones simbióticas entre especies, generalmente se suscitan de forma distribuida, por lo que cada ser tiene sus propias metas y ambiciones, pero sabe que debe acoplarse a otro ser para subsistir, o para obtener un beneficio.

También existen tres formas de realizar las tareas de acuerdo a la naturaleza e importancia de las mismas:

- Seriales
- Paralelas

Las tareas seriales son las que deben realizarse una tras otra, por lo que los agentes deben focalizarse en una misma actividad para poder avanzar en los objetivos del sistema.

En el caso de las tareas paralelas, algo más natural dentro de un sistema multi agente, permite que un agente o grupo de agentes se concentre en realizar una tarea, mientras que otro agente o grupo puede ejecutar otra tarea sin interferirse entre ellos.

Los controles cooperativos en los sistemas multi agentes están guiados por una serie de conceptos conocidos como *reglas de Reynolds* [17]. Las reglas de Reynolds capturan muy bien el movimiento colectivo de los grupos animales describiendo las interacciones entre ellos por medio de los vecinos próximos.

Para modelar un sistema multi agente que obedezca las reglas de Reynolds, generalmente se utiliza un grafo de comunicación, en donde cada nodo representa a un agente, y las aristas representan los enlaces y vecindades que se forman dentro de la *sociedad de agentes*. Se puede apreciar un ejemplo simple en la Figura 2.6 a continuación.

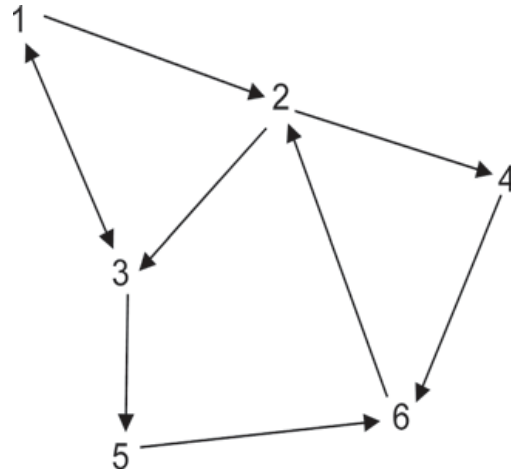


Figura 2.6. Grafo dirigido que modela el flujo de información entre un grupo social

De manera dinámica, y en un sistema de naturaleza robótica, generalmente un grafo representa distancias vecinales, lo que conlleva la obligación de actualizarlo mientras los agentes se desplacen en su medio. Los valores necesarios a actualizarse son entonces aquellos que se encuentran en las aristas. La estructura del grafo debe conservarse de manera estática debido a que es la formación deseada.

2.5 ENFOQUES EN LOS SISTEMAS MULTI ROBOT

Aterrizando todos los conceptos a agentes robóticos, es claro que la mayor necesidad de un agente es desplazarse físicamente a través de un plano espacial, que puede ser en R^2 o en R^3 . Por lo que ahora vamos a tocar temas y enfoques más específicos a este tipo de tareas.

2.5.1 Robótica basada en comportamientos

Mataric [19] nos dice que la robótica basada en comportamientos diseña controladores para dotar a los robots con comportamientos “inteligentes”, basados en “una filosofía inspirada biológicamente que favorece las arquitecturas paralelas y descentralizadas”. Propone la idea de proveer a los robots con un rango de comportamientos básicos y dejando que el entorno determine cual comportamiento es más apto como una respuesta a un cierto estímulo.

Sukhatme y Mataric [20] definen comportamientos como “procesos en tiempo real que toman entradas de sensores y/o otras percepciones, y envían salidas a actuadores y/o otros elementos físicos móviles”.

En la robótica basada en comportamientos, los comportamientos básicos son unidades fundamentales para el control, razonamiento y aprendizaje, en caso de requerirlos. El entorno juega un rol central activando ciertos comportamientos básicos en un tiempo dado. En la Figura 2.7 se aprecia el cambio de trayectoria que ejecuta un robot al encontrar un obstáculo, un claro ejemplo de influencia en un agente por su entorno.

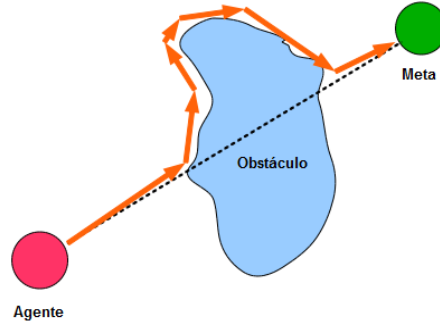


Figura 2.7. Agente modificando su comportamiento por su entorno

Generalmente, los módulos de comportamiento y los mecanismos de coordinación son diseñados a través de un proceso de prueba y error en el cual el diseñador los cambia progresivamente y los prueba continuamente, resultando en un patrón de comportamientos. Extendiendo esta clase de diseños en un sistema multi agente, esto nos lleva a un sistema completamente distribuido sin un control centralizado.

Este enfoque basado en comportamientos ha sido (y es) muy popular debido a su simplicidad en el diseño y a la adaptación fácil con entornos de la vida real. Basado en las arquitecturas supuestas sobre comportamientos de Brooks [2], Parker desarrolla la arquitectura ALLIANCE [21] para controlar grupos de robots heterogéneos y la demostró utilizando un grupo de cuatro robots físicos realizando tareas simples como empujar cajas. Divide las tareas en sub tareas dirigidas a ciertos grupos de comportamiento. Además de las condiciones basadas en sensores, Parker agrega impaciencia y aquiescencia. La impaciencia incrementa si no hay otros robots que pretendan realizar alguna sub tarea, mientras que la aquiescencia inhibe el comportamiento de un robot si no es capaz de realizar la sub tarea. Esta combinación de elementos hace que los robots cooperen de manera inherente para solucionar una tarea grande.

2.5.2 Robótica colectiva

Las colonias de hormigas, mencionadas anteriormente, son capaces de coleccionar objetos (tales como comida u otras hormigas muertas) y colocarlas en lugares particulares. El término 'comportamiento colectivo' se refiere a "cualquier comportamiento de un sistema teniendo más de un agente". Los comportamientos colectivos ofrecen la posibilidad de rendimiento en tareas mejorado, confiabilidad en las tareas incrementada, y costos menores con respecto a los sistemas robóticos tradicionales.

Mucho del trabajo realizado en la robótica colectiva se concentra en casos limitados, los cuales incluyen comportamientos de 'caza de alimento'. El término caza de alimento (flocking) se refiere a la búsqueda de varios seres por recursos para alimentarse. En el momento en el que se haya algún recurso consumible, se avisa al enjambre o colonia para que poco a poco se reúnan a recolectar colectivamente el recurso.

Los agentes típicos conformando estos sistemas en la literatura usan controladores contruidos manualmente (sin aprendizaje), o pueden realizar aprendizaje en simulación, o con dominios relativamente simples sobre entornos.

Una manera de generar comportamientos colectivos robustos es aplicar algoritmos adaptativos bio-inspirados a nivel de equipo. Se cree que la integración de métodos de aprendizaje puede contribuir significativamente al diseño de un equipo de robots auto-programables para algunas tareas predefinidas. Tal es el caso de CEBOT [9].

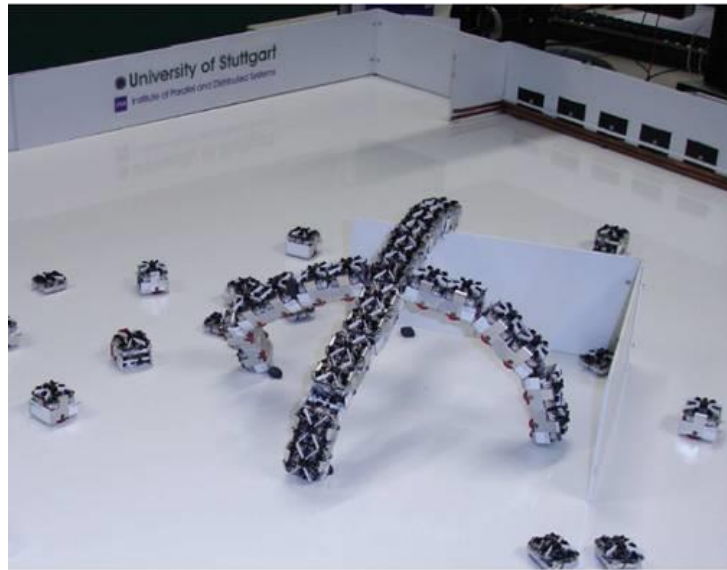


Figura 2.8. Comportamientos colectivos con robots de bajo costo

Las técnicas bio-inspiradas no son nuevas, ya que en sistemas de un solo robot, se utilizaba aprendizaje por refuerzos y algoritmos genéticos para lograr comportamientos adaptativos. En el caso de las aplicaciones multi robot, donde las funciones de *fitness* se miden a nivel de equipo, los robots se encontraran con un problema de asignación de créditos, decidiendo a qué nivel sus comportamientos han contribuido al puntaje general del equipo [22].

Existen dos maneras propuestas de lidiar con este problema. La primera es integrar la comunicación global entre los integrantes del equipo [Par95]. De cualquier forma, esta no es una solución completamente descentralizada y no entra en la definición de la robótica bio-inspirada. Además, dependiendo de las condiciones del entorno, la comunicación global no siempre es posible y tiende a generar cuellos de botella cuando el tamaño de los equipos incrementa.

La segunda manera es medir el rendimiento individual para cada robot, en vez de hacerlo como equipo [Mat96]. Aunque es una aproximación más simple, algo importante es que este esquema fuerza el comportamiento colectivo a ser la suma de los comportamientos individuales, que precisamente no es la estrategia óptima para la resolución de problemas compartidos.

2.5.3 Robótica Evolutiva

La computación evolutiva ha atraído la atención de varios campos de investigación como una manera de resolver problemas de optimización [18]. La robótica es uno de dichos campos en el cual los investigadores han encontrado muchos rangos de aplicación que van desde la síntesis de estrategias de control a *motion planning* geométrico. Como resultado, una nueva ola de investigación en la robótica comenzó a emerger; podemos llamarla robótica evolutiva.

En la robótica evolutiva, hay un número de problemas representativos de esta área; algunos están relacionados con el diseño de los robots, mientras otros están relacionados con la planeación y el control.

Cuando hablamos de computación evolutiva, generalmente nos referimos a problemas que se resuelven con aproximaciones pertenecientes al cómputo suave. El cómputo suave hace referencia a técnicas que abordan las situaciones de maneras diferentes a reglas de programación estrictas como reglas de producción, ya que permiten la retroalimentación y la adaptación. Algunas de las técnicas típicas de la computación evolutiva son los algoritmos genéticos, o algo más representativo, las redes neuronales artificiales. La Figura 2.9 nos da una representación visual de una red neuronal artificial.

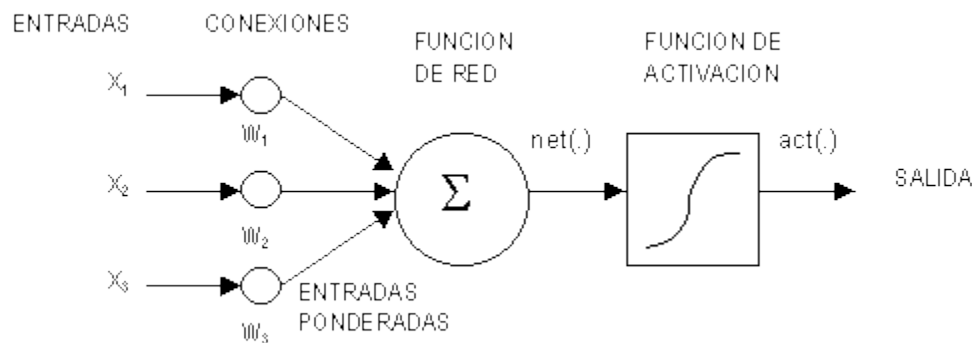


Figura 2.9. Estructura gráfica de una red neuronal artificial

La robótica evolutiva es un enfoque nuevo y prometedor al desarrollo de sistemas multi robot capaces de reaccionar de manera rápida y robusta en entornos tanto reales como simulados. En esta disciplina, se utilizan algoritmos mayormente inspirados por la evolución biológica para el diseño de sistemas de control con sensores.

A pesar de que en la época temprana, la evolución artificial fue vista principalmente como una estrategia para desarrollar controladores robóticos más complejos, hoy en día el campo se ha vuelto mucho más sofisticado y diversificado. Floreano y Urzelai [23] han identificado al menos tres enfoques en la robótica evolutiva: *ingeniería automatizada*, *vida artificial*, y *psicología/biología sintética*. Estas tres aproximaciones se traslapan entre ellas en cuanto a conceptos, pero aun así persiguen diferentes metas que eventualmente se diferencian en los resultados obtenidos.

La *ingeniería automatizada* es "sobre la aplicación de la evolución artificial para el desarrollo automático de algoritmos y máquinas mostrando habilidades complejas que son difíciles de

programar con técnicas convencionales". Mucho de ello se refiere a que el mismo diseño y funcionamiento de los robots puede variar en cada ejecución de tareas, sobre todo si los robots son auto configurables.

La *vida artificial* trata sobre "la evolución de criaturas artificiales que muestran propiedades vivientes". Esto se refiere más a robots con capacidades de aprendizaje artificial, mas no modifican totalmente su estructura para adaptarse.

La *psicología/biología sintética* intenta "entender el funcionamiento de los mecanismos biológicos y psicológicos a través de la evolución de dichos mecanismos en un robot puesto en condiciones similares a aquellas de los animales que se esté estudiando". Se trata más de emular comportamientos o características de adaptación animal.

Como ya se había mencionado anteriormente, las redes neuronales artificiales predominan como las técnicas utilizadas en los sistemas de clase predominante en la robótica evolutiva. Las redes neuronales artificiales se pueden ver como nodos simples conectados juntos por conexiones dirigidas por las cuales fluyen señales. Los nodos realizan un mapeo de entrada-salida que es usualmente algún tipo de función *sigmoide*.

La gente que ha aplicado esta técnica generalmente establece los siguientes criterios sobre por qué las redes neuronales artificiales son preferidas:

1. Variando las propiedades y los parámetros de las unidades simples de procesamiento utilizadas, diferentes tipos de funcionalidad pueden alcanzarse con el mismo tipo de estructura de red. Esto significa que los mismos esquemas de codificación pueden utilizarse independientemente de la funcionalidad del sistema de control.
2. Utilizando redes neuronales artificiales permite implementar y probar ideas de la biología acerca de cómo los mecanismos neuronales pueden funcionar para la generación de comportamientos. También, permiten probar y refinar los modelos biológicos propuestos a través de la exploración del espacio de parámetros que pueden sugerir nuevas hipótesis.
3. Existen otros tipos de procesos adaptativos que se pueden emplear en conjunto con la evolución artificial, tales como varias formas de aprendizaje supervisado y no supervisado.
4. Los comportamientos en los que la robótica evolutiva está inmiscuida hasta el momento son los comportamientos a bajo nivel, altamente emparejados con el entorno a través de ciclos de retroalimentación simples y precisos. Las redes neuronales artificiales son ideales para este propósito, dada su naturaleza.

Mucho del trabajo de la robótica evolutiva que ha sido explorado concierne a la evolución de redes neuronales de arquitectura fija para el control de robots. Incluso, también es posible evolucionar las redes neuronales para controles robóticos cuyo tamaño y morfología están bajo el control de un proceso evolutivo por sí mismas.

2.6 PROBLEMAS INHERENTES EN UN SISTEMA MULTI ROBOT

Ya se han abordado temas sobre las características que deben cumplir los robots para pertenecer a un sistema multi robot, así también como la teoría relativa a los sistemas multi agentes en general. En algunas partes del trabajo se mencionaron interferencias en las actividades de los robots cuando conviven en el mismo entorno, además de mencionarse que es necesario obtener y ofrecer información al equipo para que cada agente esté al tanto de las actividades que debe realizar, así también para informar si es o no capaz de realizar alguna tarea.

Una de las principales interferencias que se conciben en los sistemas multi robot es la modificación constante del entorno. Y es que aunque un robot tiene compañeros de equipo que pueden ayudar en las tareas globales, también cada uno de ellos ocupa un espacio físico, lo que conlleva a tener más obstáculos que no estaban contemplados en un inicio, y lo peor, dichos obstáculos se mueven de una manera no planificada en cada instante de tiempo. Esto nos limita en cuestiones de planificación, ya que no se puede tener una planificación completamente estática, puesto que las trayectorias pueden verse interferidas debido al bloqueo instantáneo de otros elementos, y en el peor de los casos, terminar con caminos sin salida.

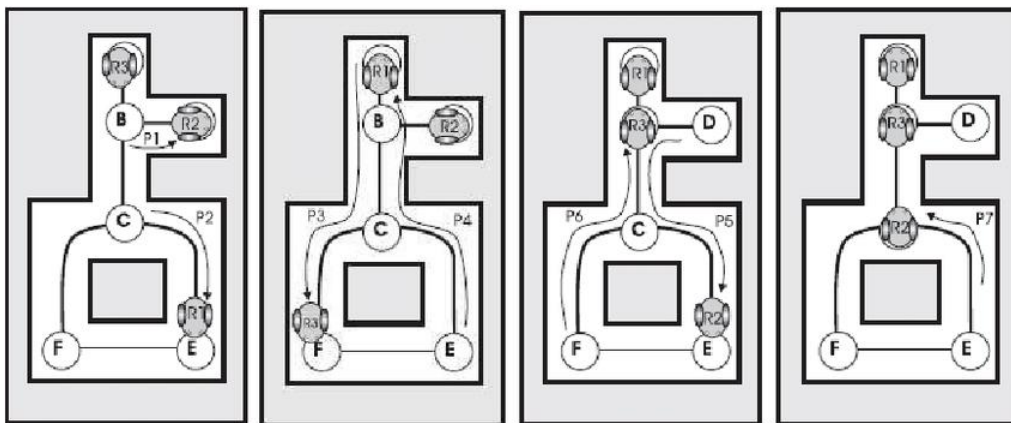


Figura 2.10. Problema de path planning simultáneo

Naturalmente, si este problema se presenta, quiere decir que necesitamos desarrollar un esquema de organización dentro del grupo de robots para minimizar o evitar posibles interferencias. Esto implica que necesitamos algunos mecanismos adicionales, que quizá se hayan mencionado antes, pero esta vez se agrupan en forma de lista:

- Comunicación
- Protocolos de asignación de tareas
- Coordinación y planificación conjunta

Cabe destacar que los tres requerimientos antes mencionados dependen en implementación de acuerdo a las características de los agentes en el sistema que se desarrolle. No es lo mismo planear acciones conjuntas para agentes cuya naturaleza es cooperativa en primera instancia, contra otros agentes cuya naturaleza sea competitiva, puesto que en la segunda,

los agentes buscarán la manera de impedir que otros agentes cumplan sus objetivos o intentar cumplir los suyos a expensas de los demás.

En un pequeño resumen sobre los sistemas multi agentes, debe tener las siguientes propiedades:

- Autonomía: los agentes tienen algún estado, y toman decisiones sobre qué hacer basándose en dicho estado.
- Capacidad de reacción: los agentes están situados en un entorno, que puede ser el mundo real, un usuario a través de una interfaz gráfica de usuario, Internet, o una combinación de los anteriores y son capaces de percibir este entorno, y responder de manera oportuna a los cambios que ocurren en él.
- Pro-actividad: los agentes no actúan simplemente en respuesta a su entorno, sino que pueden exhibir un comportamiento dirigido a objetivos tomando la iniciativa.
- Sociabilidad: los agentes interactúan con otros agentes mediante algún tipo de lenguaje de comunicación de agentes, y típicamente poseen la habilidad de comprometerse en actividades sociales como resolución cooperativa de problemas o negociación para lograr sus objetivos.

Existen dos tipos de robots agentes, deliberativos y reactivos. Los robots deliberativos adquieren el conocimiento por patrones, se mueven en un entorno simbólico, y muestran interacción con otros grupos de agentes, deciden la acción a realizar teniendo en cuenta su historia de interacciones con el entorno, la secuencia de estados con el entorno o de percepciones. Los robots reactivos no tienen ningún patrón, tienen capacidades sensoriales y motrices, y actúan por reflejos y sin planificación. Deciden la acción a realizar teniendo en cuenta el estado actual del entorno, sin tener en cuenta su historia.

Bajo ciertos criterios, un robot reactivo es más fácil de programar, pero ligeramente más difícil de coordinar con otros agentes, aunque es el tipo de agentes que reaccionan de una manera más rápida a los cambios de estrategias y planificación de movimientos [4].

3 ARQUITECTURAS PARA LA COORDINACIÓN MULTI ROBOT

La coordinación multi robot [24] involucra el uso de mecanismos para robots que puedan tener diferentes habilidades para cooperar y llevar a cabo tareas globales. La coordinación multi robot está directamente relacionada a la coordinación multi agente, en la cual el los agentes de software cooperan tomando en cuenta las acciones hechas por los demás agentes.

La coordinación en el campo de la robótica tiene que lidiar con problemas relevantes tales como el manejo de la incertidumbre, problemas de comunicación físicos, y restricciones físicas de los robots en entornos altamente dinámicos.

Una línea relevante de investigación en robótica es cómo manejar la coordinación entre robots compartiendo una meta en común en entornos dinámicos. Para confrontar el asunto, los sistemas multi agente [25] encajan perfectamente en este problema, como ya se ha visto en el capítulo anterior, lo cual involucra fuentes heterogéneas de conocimiento que cooperan para alcanzar una meta en común.

Por otro lado, un protocolo de coordinación puede ser usado para que los robots interactúen entre ellos dentro de su entorno para hacer decisiones en la dirección correcta. La idea consiste en definir un conjunto de reglas que los robots deben seguir para alcanzar objetivos parciales como la asignación de roles, la distribución de tareas, o incluso la exploración del entorno [26].

Para encarar el problema de la implementación de sistemas multi robot, se puede hacer uso de paradigmas de sistemas multi agente para controlar los robots. Específicamente, es de particular interés en cómo se establecen dinámicamente equipos de agentes para coordinarlos cuando se obtiene información de su entorno. Esto es particularmente útil en entornos donde los robots son de tipo homogéneo, es decir, que todos los robots son iguales.

La primera etapa, independientemente de la clasificación y la taxonomía del sistema en general, es establecer un protocolo de comunicación, el cual debe englobar diversos aspectos:

- Medios físicos de comunicación
- Establecer un mismo idioma para todos los robots
- Elegir modelos de comunicación de acuerdo al tipo de sistema y sus capacidades

Además de los tres puntos arriba descritos, es necesario establecer algunas reglas en la comunicación, que sí están ligadas a la taxonomía y naturaleza del sistema. Existen dos conjuntos de reglas del que se pueden tomar conceptos para aplicarlos:

- Comunicación directa
- Comunicación indirecta
- Comunicación nula

Incluso, dependiendo del protocolo de los medios físicos de comunicación, además de las decisiones que se hayan tomado para el sistema, es posible elegir entre dos tipos de atención a los mensajes que se mandan en la comunicación. Los dos tipos son:

- Comunicación síncrona
- Comunicación asíncrona

Los modelos de comunicación generalmente van ligados al fin que se quiere alcanzar en el sistema, además de los recursos con los que se cuentan. De acuerdo a la naturaleza de los agentes, es necesario elegir el modelo adecuado para que entre ellos se facilite la comunicación, tomando en cuenta la disminución de interferencias y pérdidas en la transmisión de los mensajes, ya que generalmente los robots que forman parte de sistemas multi robot comunicados, efectúan el intercambio de información a través de módulos inalámbricos.

Dejando por el momento de lado el tema de la comunicación, ahora se debe elegir las diferentes técnicas de coordinación. Principalmente se tienen tres enfoques en un sistema multi robot, del cual se desprenden diversas técnicas para alcanzar los objetivos deseados.

Los enfoques en los que podemos clasificar la coordinación multi robot son:

- Planificación centralizada
- Planificación distribuida
- Planificación centralizada en planes parciales (mixta)

Cada una de ellas se verá más a fondo a lo largo del capítulo. Por lo mientras, cabe destacar que cada una de ellas tiene un conjunto de estrategias, excepto por la tercera que es una mezcla de las dos anteriores, de manera ordenada.

Recordamos que los agentes robóticos tienen comportamientos diferentes, los cuales se pueden listar en:

- Apatía
- Cooperación
- Competencia

Mientras que si nosotros clasificamos a los agentes robóticos por el tipo de planificación que tienen, podemos encontrar dos tipos:

- Agentes robóticos deliberativos
- Agentes robóticos reactivos

Otro elemento importante, que va más ligado a la composición física de los robots son las capacidades motoras y sensoriales de cada individuo, además de las dimensiones que ocupa. Los robots se clasifican en tres tipos esenciales:

- Robots Estáticos
- Robots Híbridos
- Robots Móviles

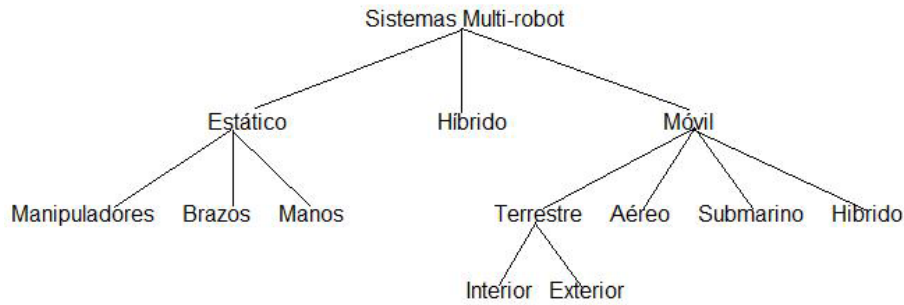


Figura 3.1. Clasificación de los sistemas multi robot [4]

En la Figura 3.1 se aprecian diferentes ejemplos para cada tipo de robot. Un sistema puede estar conformado de un mismo tipo de robots, conocidos como sistemas homogéneos; o puede estar conformado por dos o más tipos de robots, lo cual se conoce como sistemas heterogéneos. Particularmente, los sistemas de robots móviles terrestres y los formados por manipuladores, son los sistemas multi-robot más estudiados; los primeros se denominan enjambres (*swarms*), colonias (*colonies*), o colectivos (*collectives*), lo que indica que la Biología y la Etología han sido la base inspiradora de muchos de estos desarrollos, como ya se vió en el capítulo anterior.

En este capítulo analizaremos a fondo todos los detalles que se presentaron en esta breve introducción, así también como aspectos adicionales que se deben tomar en cuenta como parte de una arquitectura para la coordinación de múltiples robots.

Es muy importante resaltar que las arquitecturas están orientadas a cumplir específicamente dos objetivos.

1. Cubrir los requerimientos para la coordinación de un esquema fijo de robots, realizando múltiples tareas.
2. Cubrir los requerimientos para la coordinación de un esquema fijo de tareas, utilizando diferentes tipos de robots.

Con lo anterior, se destaca entonces que tenemos arquitecturas orientadas a tareas y arquitecturas orientadas a grupos de robots específicos. No obstante, esto no significa que el número de tareas (o robots en su caso) sea fijo, lo que es fijo es la naturaleza de cada elemento, según sea el esquema.

El primer tema a tocar son los modelos de comunicación, que independientemente de los robots que se vayan a utilizar o las tareas a realizar, es completamente necesario establecer el intercambio de mensajes, con la finalidad de saber si se ha alcanzado el objetivo general, o informar el estado en el que se encuentra cada agente robótico.

3.1 PROTOCOLOS Y MODELOS DE COMUNICACIÓN PARA LA COORDINACIÓN

Cada agente debe ser capaz de comunicarse con otros agentes, de forma que pueda provocarles cambios en sus objetivos o creencias, cuando esto sucede se dice que dos agentes han establecido un acto comunicativo, al agente que toma la iniciativa de la comunicación

lo llamaremos hablante y al agente al que va dirigida dicha acción oyente. Existen varias categorías de actos de habla, tales son los representativos, los directivos, los exhortativos, los expresivos y los declarativos. En la Tabla 3.1 se expresan las descripciones de cada acto de habla.

Acto de habla	Descripción
Representativos	Exponen una determinada proposición como representación de un estado de cosas del mundo.
Directivos	Mediante los que se realiza una petición, se da una orden, en éstos se intenta que el oyente actúe de tal modo que su conducta concuerde con el propósito.
Exhortativos	Para comprometer, amenazar, constituye un compromiso por parte del hablante de adoptar el tipo de acción representada en el contenido proposicional.
Expresivos	Para expresar un agradecimiento, felicitaciones, bienvenidas o pedir disculpas.
Declarativos	Actos que producen un cambio en el estado del mundo

Tabla 3.1. Categorías de actos de habla

La comunicación forma la base de la coordinación, existen diferentes métodos de comunicación el caso más simple es la invocación a procedimiento de un agente por otro agente.

Los protocolos de comunicación se especifican en diferentes niveles. El nivel inferior del protocolo especifica el método de interconexión; el nivel medio especifica el formato o sintaxis de la información que es transferida; el nivel superior especifica el significado o semántica de la información. La semántica se refiere tanto a la esencia del mensaje, como al tipo de mensaje. Existen protocolos de comunicación binarios y n-arios. Un protocolo binario involucra a un único emisor y aun único receptor, mientras que un protocolo n-ario involucra a un único emisor y a múltiples receptores.

Un protocolo está especificado por una estructura de datos con los siguientes cinco campos: emisor, receptor, lenguaje en el protocolo, funciones de codificación o decodificación, acciones realizadas por el receptor o los receptores.

Los protocolos a los que se hacen referencia aquí son los especificados por los diseñadores de los sistemas multi robot para la comunicación de los agentes que lo conforman. Es entonces que cada protocolo debe considerar diseños a la medida de las necesidades, aunque en la práctica existen diversas configuraciones.

En un ejemplo, supongamos que un robot desea dar a conocer su posición a los demás, por lo que, en el mensaje que envíe, el robot debe especificar algunos datos sobre quién es el emisor, además de la posición geo-espacial. Este tipo de mensaje forma parte de un acto de habla representativo, puesto que está dando a informar la situación actual de un elemento del entorno, en este caso él mismo.

Por lo tanto, podemos considerar el siguiente ejemplo de mensaje compuesto de varios campos:

Nombre de campo	Tipo de dato	Descripción
Emisor	String	Campo descriptivo que permite identificar al robot de los demás, puede ser un ID establecido por el diseñador.
Tipo de mensaje	Int	Un identificador para saber qué tipo de mensaje se trata.
Posición	Point 3D	Campo que contiene un punto en 3D, que a su vez contiene tres valores espaciales (x,y,z).
Tiempo	Time	La hora/ fecha en la que se envía el mensaje.

Tabla 3.2. Ejemplo de mensaje con una posición espacial

En la Tabla 3.2 se muestra con detalle la composición de un mensaje de posición espacial de un robot. El campo ‘tipo de mensaje’ es muy importante, puesto que les indica a todos los receptores la manera en la que se debe procesar el mensaje. Independientemente de la información que se esté enviando, es de suma importancia incluir la fuente del mensaje y algún tipo de información acerca del contenido.

La estructura de los mensajes que mandan los agentes robóticos entre sí, forma parte de la sintaxis del lenguaje que se diseña para el robot. Otros elementos que deben conformar el lenguaje del sistema son protocolos de comunicación física (como en las primeras capas del modelo OSI), protocolos de comunicación a medio nivel (tecnologías como sockets TCP/IP o UDP son incluidas aquí), hasta protocolos de alto nivel en capas de aplicación (http, por ejemplo).

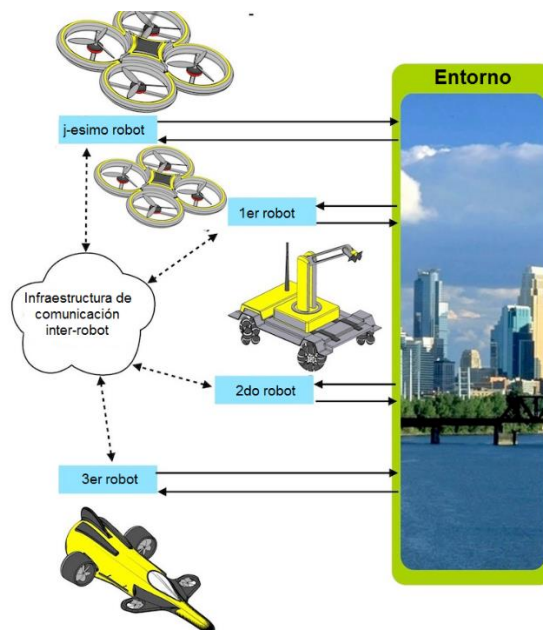


Figura 3.2. Modelo de interacción entre múltiples robots

Una vez que se establecen los puntos arriba señalados, es importante conocer la estructura de comunicación que van a formar los agentes robóticos. Si recordamos parte del capítulo anterior, utilizamos un grafo para describir las interconexiones existentes. A este tipo de enfoque, en los sistemas multi robot se le conoce como *rango de comunicación limitado*, que se refiere a tener a un subconjunto del conjunto total de agentes con los que cada robot puede establecer comunicación. Existe otro enfoque llamado *rango de comunicación ilimitado*, el cual define que cada robot puede establecer comunicación con todos los agentes por medio de diversas técnicas

3.1.1 Rango de comunicación limitado

En este enfoque, consideramos que cada robot tiene un rango de comunicación circular con otros robots, cuyo radio es denominado P , es entonces cuando un robot considera a sus vecinos cercanos dentro del rango R que es igual a:

$$R = \pi * P^2$$

Una vez que los tiene identificados, generalmente [4], los robots comienzan a intercambiar información sobre sí mismos para identificarse como vecinos, por lo que es importante que cada robot cuente con su propia estructura de datos que almacene los contactos que va creando en cada momento.

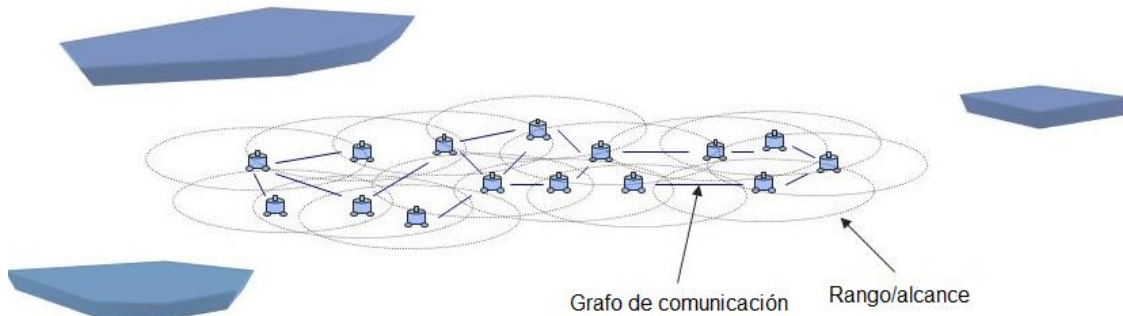


Figura 3.3. Ejemplo de rango de comunicación limitado

En la Figura 3.3 se aprecia la configuración del grafo que se forma de acuerdo al alcance del rango que tiene cada robot individualmente, todos contando con el mismo rango P . Es importante remarcar las siguientes observaciones dentro de este esquema:

- El rango no necesariamente aplica en un área plana, generalmente se distribuye en un entorno tridimensional, sin embargo, si todos los robots se mueven únicamente con dos coordenadas y todos se encuentran a la misma altura, se puede simplificar de esta forma.
- Es posible que los rangos varíen en la práctica, por lo que se utiliza un promedio de las muestras de cada robot para determinar el rango para todos los robots.
- El rango debe ser determinado por especie de robot, es posible contar con agentes robóticos equipados con mejores o peores aditamentos para establecer comunicación

(generalmente inalámbrica) por lo que se debe considerar que ambos robots puedan abarcarse mutuamente con sus rangos para generar una relación entre ellos.

Uno de los grandes problemas que tiene este esquema es que requiere tener procesos activos en cada uno de los agentes para poder percibir y estimar a quién tiene dentro del rango de percepción. Esto implica un gasto importante de recursos; sobre todo de batería, dado que la autonomía de los robots generalmente se da a través de estos mecanismos de energía.

Otro de los problemas es poder mapear completamente el grafo de comunicaciones, ya que la naturaleza de un sistema con un rango estrictamente limitado implica que todos los agentes deben compartir la información que poseen para ir estructurando el grafo. Si contamos que el grafo debe actualizarse cada cierto tiempo debido al movimiento de los robots y las modificaciones de las posiciones de los agentes vecinos, puede resultar casi imposible mapear completamente el grafo.

Este tipo de enfoque sirve más para sistemas compuestos por robots estáticos o robots que no varíen mucho su posición, dado un rango, e igual procurando mantener la estructura de vecindades en el grafo.

Existen dos enfoques en el manejo de la comunicación para el rango limitado. El primero es un protocolo 'sin mensajes', aunque realmente no se deja de lado la comunicación, los robots únicamente informan datos efímeros para evitar las colisiones, sin embargo, no se comunican intenciones ni objetivos. Este modelo es más simple puesto que el robot únicamente comunica su posición a los que pueda alcanzar para que los demás ubiquen un obstáculo en sus mapas internos.

El segundo enfoque es un protocolo basado enteramente en mensajes. En este enfoque, el robot comunica a todos los que tenga alcance toda la información que posea y pueda ser de utilidad para los demás agentes. En un sistema de robots móviles, por ejemplo, lo que es posible comunicar son los árboles de exploración que cada robot haya creado para evitar explorar partes del entorno.

Ambos enfoques utilizan lo que se conoce formalmente como comunicación directa, ya que los elementos envían mensajes sin ningún tipo de intermediario, y en caso de solicitar información adicional, los robots pueden establecer una especie de 'diálogo' para coordinar acciones.

3.1.1.1 Comunicación directa (Sistemas mensaje-diálogo)

Como se había mencionado anteriormente, el mecanismo de comunicación directa (mensaje-diálogo) funciona a través del intercambio directo de mensajes entre agentes. Lo esencial aquí es que se imita un modelo de comunicación típica social.

Como se ha expuesto en la sección 3.1, los mensajes deben tener cierto tipo de estructura que permita a los agentes conocer el contenido de los mismos para poder interpretarlos de manera adecuada. Generalmente un diálogo no es lo mismo que simplemente informar algunos datos a otros agentes sin que los hayan solicitado anteriormente.

El diálogo es una forma de comunicación verbal o escrita en la que se comunican dos o más personas en un intercambio de información, alternándose el papel de emisor y receptor. Siguiendo esta definición, en un diálogo hay que tener en cuenta que es necesario un intercambio de papeles entre los denominados interlocutores. Este cambio de papeles entre emisor y receptor se denomina turnos de palabra o intervenciones.

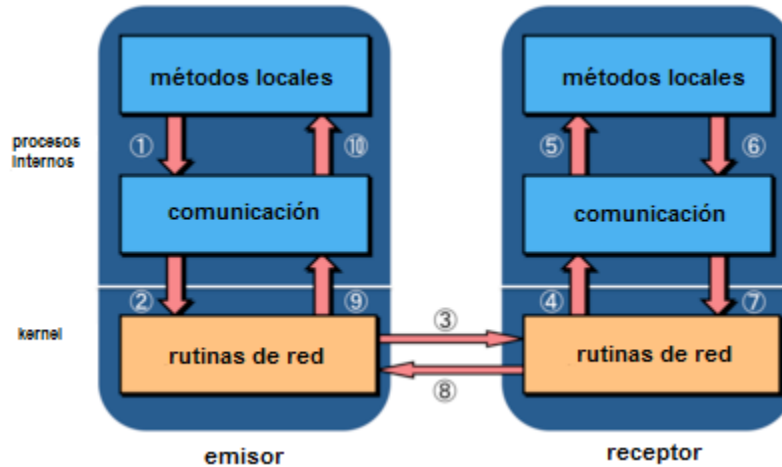


Figura 3.4. Proceso de comunicación directa emisor-receptor

Un agente toma el papel de emisor cuando es el que envía el mensaje dentro de un diálogo. El agente toma el papel de receptor cuando es quién recibe un mensaje. En la Figura 3.4 se ilustra un proceso técnico de comunicación directa, a través de diferentes funciones de los robots, con el fin de invocar un procedimiento remoto en otro robot, en caso de así requerirlo.

Es importante destacar que los robots pueden utilizar cualquier protocolo en una o más capas que deseen, siempre y cuando los participantes en el diálogo puedan interpretar las señales, para que exista el intercambio de mensajes.

3.1.2 Rango de comunicación ilimitado

En este enfoque, todos los robots tienen acceso al resto de los agentes para el intercambio de información, es decir que es una comunicación de todos con todos. En teoría, el sistema ideal permite que todos los robots puedan mandar mensajes a cada uno de los participantes en el entorno.

Por principio, este enfoque parece ser el más adecuado para la coordinación de todo el conjunto de agentes robóticos en un entorno, sin embargo, sabemos que las capacidades físicas en la comunicación no siempre lo permiten, ya que tendríamos que invertir en equipo de largo alcance para poder cubrir el espacio necesario, además de que aun así nos topamos que el rango 'ilimitado' únicamente funciona dentro de cierta extensión del entorno, por lo que realmente no es completamente ilimitado.

Tenemos dos opciones en el caso de querer implementar un sistema con rango de comunicación ilimitado:

- Lograr la comunicación completamente distribuida utilizando medios de comunicación de amplio alcance.
- Lograr la comunicación completamente distribuida a través de la replicación de la información entre agentes cercanos (enfoque mixto entre limitado e ilimitado)
- Lograr la comunicación centralizada a través de un agente principal donde todos puedan consultar mensajes.

El primer caso es el utópico, ya que no requiere gran ciencia. La forma simplificada de comunicación aquí se puede considerar a través de *broadcasting*, lo cual consiste en simplemente difundir un mensaje a través de toda la red de agentes, para que los interesados los escuchen y descifren.

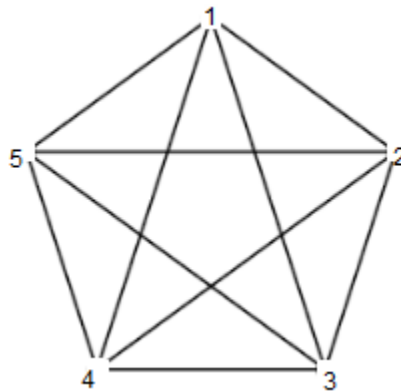


Figura 3.5. Grafo de comunicación no dirigido completo (primer caso)

El segundo caso es una aproximación del primero, ya que se pretende lograr un *broadcasting*, sin embargo con réplica de información entre los agentes que no pueden escuchar el mensaje directamente del emisor original.

En ambos casos, radican problemas de suma importancia: la saturación de los canales de comunicación, la dificultad de discernir entre mensajes útiles y mensajes sin relevancia para cada agente en particular, el tiempo de procesamiento para la cola de mensajes recibidos, la falta de tolerancia de errores, entre otros problemas particulares [27].

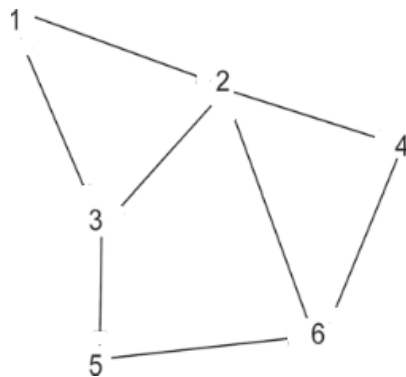


Figura 3.6. Grafo de comunicación no dirigido fuertemente conexo (segundo caso)

Tal como lo ilustran las Figuras 22 y 23, podemos contar con dos grafos de comunicación representando las conexiones en el sistema. En el primer caso se cuenta con un grafo completo, ya que todos los agentes tienen comunicación con el resto del grupo. En el segundo caso, se cuenta con un grafo fuertemente conexo que permite la comunicación a todos los agentes a través de caminos.

El tercer caso trata de aliviar varios de los problemas que los otros enfoques presentan, que es el caso de tener una fuente de información centralizada, conocida como *boletín*. En ella los agentes pueden publicar y consultar mensajes. Toda la información que se procesa debe estar controlada por un agente exclusivo a la tarea, el cual es el centro de la comunicación, por ello se le conoce como un tipo de comunicación centralizada.

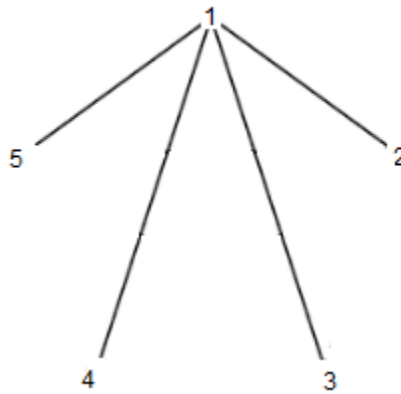


Figura 3.7. Grafo de comunicación no dirigido fuertemente conexo con un centro de comunicación (tercer caso)

Es posible implementar un sistema de comunicación central sin supervisión, sin embargo, en la práctica puede ser perjudicial en cuanto a tiempos y recursos, ya que cada agente debe tomarse el tiempo de estructurar los mensajes que los otros agentes dejaron en dicho centro.

Para cualquiera de los casos, este enfoque de rango de comunicación ilimitada utiliza técnicas denominadas *comunicación indirecta*, puesto que los mensajes no tienen un destinatario fijo o único, o dicho mensaje debe atravesar varios destinatarios antes de poder llegar al receptor final.

3.1.2.1 Comunicación indirecta (Sistemas de pizarras y broadcasting)

La comunicación indirecta hace referencia a todos los tipos de comunicación que no tratan de entablar un diálogo directo entre dos entidades (emisor-receptor), sino que establecen un intercambio de mensajes en el que varios (o todos) los agentes de un sistema pueden recibir mensajes.

La primera aproximación, conocida como *broadcasting* es un concepto nativo de las redes de computadoras. Dentro de una red local, cada computadora está identificada a través de una dirección IP (Internet Protocol), lo que permite mandar mensajes a destinatarios concretos. En las redes, existen dos direcciones reservadas del rango de direcciones posibles a asignar, una de ellas es la dirección utilizada para enviar mensajes broadcast. Técnicamente, lo que

hace la infraestructura de la red es replicar el mensaje a todas las direcciones activas en la red, a través de un router. La Figura 3.8 ilustra el flujo de un mensaje broadcast a través de elementos en una red.

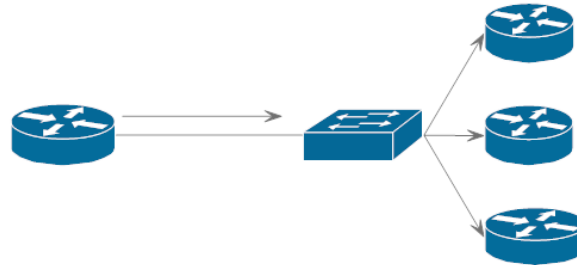


Figura 3.8. Red de broadcast

El sistema de broadcasting tiene ciertas desventajas, ya que todos los agentes están recibiendo el conjunto completo de mensajes que fluyen en la red, lo que podría ralentizar el proceso de comunicación al recibir tantas peticiones de recepción de mensajes, por lo que este tipo de esquemas sólo es viable para mensajes importantes que deban ser escuchados por el conjunto entero de agentes.

Para aliviar esta problemática, existe otro enfoque conocido como *sistema de pizarra* (del inglés *blackboard*), que consiste esencialmente en una estructura de datos compartida. Los agentes pueden usar dicha pizarra para comunicarse simplemente escribiendo y leyendo mensajes en ella.

La resolución de problemas con pizarras, toma como base la metáfora de un grupo de especialistas trabajando en conjunto para resolver un problema, utilizando para ello una gran pizarra. Una vez que el problema y los datos iniciales son escritos sobre la pizarra, cada agente podrá aportar su experiencia realizando una contribución que es escrita sobre la pizarra. La contribución de un especialista podrá habilitar a que otros agentes también realicen sus contribuciones, continuándose el proceso hasta que el problema finalmente es resuelto.

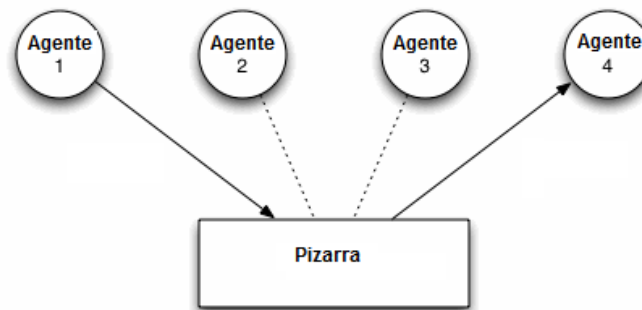


Figura 3.9. Funcionamiento de un sistema de pizarra

Generalmente una pizarra es gestionada y arbitrada por un controlador o moderador[28] que permite discernir entre los mensajes legibles, y poderlos colocar en el orden correcto.

Frecuentemente, el moderador es considerado un agente adicional (no robótico) cuyas tareas son exclusivamente la coordinación de la pizarra, así como la distribución de los mensajes.

3.2 MÉTODOS Y PROTOCOLOS PARA LA COORDINACIÓN

En los sistemas multi agente cooperativos, el diseñador puede imponer un protocolo de interacción y una estrategia para cada agente. El análisis está centrado en determinar los resultados sociales que se obtendrán dado el protocolo y asumiendo que los agentes usan las estrategias impuestas. Este tipo de escenario ha sido estudiado para la resolución de problemas distribuidos. También abarca desarrollos más recientes como problemas de juegos de equipo. Algunos de los mecanismos que han sido aplicados en estas situaciones se presentan a lo largo de esta sección [29].

Nuestro estudio se concentra sobre el problema muy general de la cooperación entre agentes situados (o robots) autónomos. Se trata de integrar esta noción en el corazón de los agentes que participan en la resolución de problemas distribuidos. Es decir, la cooperación es en ocasiones necesaria en el cumplimiento de ciertas tareas y puede ser sinónimo de aumento de desempeños:

- Realización de una tarea que requiere la combinación cuantitativa de esfuerzos de diversos agentes.
- Realización de una tarea que requiere la combinación de competencias diversas repartidas entre diversos agentes especializados (noción cualitativa).
- Prevención o resolución de conflictos (ligados al uso de un recurso)
- Ayuda mutua, repartición de tareas y compartir recursos para la mejora de los desempeños individuales o globales.

La cooperación es una de las características más interesante de los sistemas multi agente.

Es por otro lado la fuente misma del desarrollo del área de estudio. Es decir, un sistema no tiene un buen desempeño solo en el caso de que los agentes sean capaces de organizarse para producir un todo superior a la suma de los desempeños individuales.

Las formas de cooperación posibles son numerosas y dependientes del tipo de sistema multi agente considerado. Así, no existe definición general, pero más bien existen definiciones parciales o ad hoc a los diversos enfoques de la Inteligencia Artificial Distribuida [13].

Por el contrario, es posible considerar criterios de desempeño que permitan medir los efectos de la cooperación sobre un sistema (por comparación con los agentes no cooperativos). Estos son criterios cualitativos o cuantitativos concernientes a la resolución de un problema general (por ejemplo la cantidad de mineral recogido por los robots mineros en un periodo de tiempo) debido a que resultaría muy complicado cuantificar el desempeño de los robots, punto que se tocó en el capítulo anterior.

Este tipo de análisis ha permitido definir cuatro categorías de ventajas ligadas a los comportamientos cooperativos [4]:

1. Realizar tareas imposibles de manera individual,
2. Mejorar la productividad de cada uno de los agentes,
3. Disminuir el tiempo de realización de una tarea, o aumentar el número de tareas realizadas en un plazo asignado,
4. Mejorar la utilización de recursos: colaboración con el fin de obtener una de las ventajas 2 o 3.

Ahora que se va a profundizar en las técnicas de cooperación, lo que cada técnica debe ser capaz de manejar en cuanto a los agentes debe ser:

- Resolución de conflictos
- Asignación de tareas
- Manejo de la jerarquía en los agentes

Diversas técnicas requieren el manejo de roles en los agentes, para poder organizar de manera adecuada la jerarquía en el sistema. Dichos roles pueden ser estáticos (se conservan durante la ejecución del programa o son establecidos directamente), o dinámicos (cambian según sea necesario adoptarlos). Las capacidades individuales de los agentes deben ser tomadas en cuenta para una correcta asignación de roles.

Las técnicas que se van a estudiar en esta sección son:

- Asignación de tareas fija
- Reagrupamiento
- Especialización
- Arbitraje
- Negociación
- Repartición de tareas, información y recursos
- Coordinación de acciones

No todas las técnicas están disponibles para la resolución de todos los conflictos presentes en cualquier sistema, es necesario considerar los aspectos como el comportamiento de los agentes y las tareas a resolver. Recordando que tenemos diferentes comportamientos en los agentes, podemos recapitular los siguientes:

- **Agentes reactivos.** Agentes que reaccionan a su entorno de manera espontánea de acuerdo a las percepciones actuales de su entorno.
- **Agentes deliberativos.** Agentes que reaccionan tomando decisiones de acuerdo al historial de acciones que poseen.
- **Agentes vivientes.** Tipo especial de agente que simula comportamientos de la naturaleza (animales).

Los métodos de cooperación generalmente empleados se resumen en la Tabla 3. Se ha indicado para cada método el tipo de sistema (comportamiento de los agentes) que puede utilizarse y las técnicas empleadas en cuanto a la reacción de los agentes [14].

Método	Sistema	Técnica
Asignación directa	n/a	pre-programado
Reagrupamiento	reactivos/vivientes	reflejo/percepción
Especialización	reactivos/deliberativos	por el ambiente
Arbitraje	deliberativo/reactivo	pre-programado
Negociación	deliberativo	comunicación directa/indirecta
Repartición de tareas, información y recursos	deliberativo reactivo	comunicación directa/indirecta
Coordinación de acciones	deliberativo reactivo	comunicación directa/pre-programado/percepción/marcadores

Tabla 3.3. Métodos de cooperación según el tipo de sistema

Asignación directa. Es un proceso simple en el que se establecen por defecto las tareas en las que cada agente debe participar, generalmente no es necesario ningún tipo de comunicación, debido a que cada agente actúa en una orquestación predeterminada por el diseñador. Estos sistemas no tienen flexibilidad ni son capaces de tolerar fallos cuando suceden intervenciones exteriores.

Reagrupamiento y multiplicación. La simple agregación de los agentes es una forma de cooperación (esta asegura a los animales una mejor seguridad, más calor y la compartición de la comida). El reagrupamiento simplifica la navegación de numerosos agentes (vuelo de los pájaros o bancos de peces). Es necesario que un solo agente (o un pequeño grupo) decida de la dirección a tomar y todos los demás no tienen que seguirla por un comportamiento local simple [18]. La multiplicación de los agentes, es decir la redundancia de las competencias, asegura una gran fiabilidad (o robustez). En contraparte, todo reagrupamiento puede en ciertos casos críticos crear situaciones de estorbo y de conflictos.

Especialización. Es un proceso que conduce de manera progresiva a un agente a especializarse en algunas de sus tareas. Se trata de una adaptación dinámica del agente a las condiciones y a las necesidades de su ambiente (generalmente en los sistemas reactivos [30]).

Arbitraje y negociación Son dos medios para administrar los conflictos entre agentes. El arbitraje establece las reglas sobre el comportamiento de los agentes que tienen para las consecuencias en el nivel global la limitación de los conflictos. La negociación permite a los agentes cognitivos resolver los conflictos. Estos se enfrascan en una verdadera “discusión”, y necesitan un sistema de comunicación de alto nivel, con la finalidad de conseguir la obtención de una solución o un compromiso.

Repartición de las tareas, la información y los recursos Se trata de los procesos colaborativos que permiten a los agentes repartir las tareas, las informaciones y los recursos con la finalidad de realizar un objetivo común. Esta repartición puede hacerse en el seno de los sistemas deliberativos por medio de mecanismos de oferta y de demanda ([15] [31]). En los sistemas reactivos, esta repartición se hace por medio del ambiente, y conduce a la especialización de los agentes y a su repartición geográfica.

La coordinación de acciones “En el estudio de la cooperación, esta puede ser definida como la articulación de las acciones individuales cumplidas por cada uno de los agentes de manera que el conjunto desemboque en un todo coherente y adecuado”.

Son todas las tareas o acciones que no son directamente productivas pero que aseguran el cumplimiento de aquellas que sí lo son. Se distinguen cuatro formas de coordinación de acciones:

- **Coordinación por sincronización** (búsqueda de un encadenamiento coherente de acciones no paralelizables, muy presente en los sistemas automáticos industriales y los sistemas operativos distribuidos).
- **Coordinación por planificación** (necesita ya sea de un agente coordinador que planifique las acciones de cada agente, o ya sea el intercambio y la construcción de planes parciales con la idea de converger hacia un plan global [14]).
- **Coordinación reactiva** (ninguna planificación, estos son agentes reactivos que se auto organizan a través de sus interacciones con el ambiente).
- **Coordinación por reglamentación** (reglas de comportamiento que podrían evitar los conflictos).

3.2.1 Red de contratos

Del inglés *contract net protocol*, es un protocolo de interacción para la resolución de problemas entre agentes cooperativos. Se aplica principalmente a la asignación flexible de tareas entre múltiples agentes. En este protocolo, los agentes pueden dinámicamente adoptar dos roles: administrador ó contratista. Dada una tarea, un agente determina si puede dividirla en subtareas que pueden ser ejecutadas concurrentemente. Luego emplea el protocolo de red de contratos para anunciar las tareas que pueden ser transferidas, requiriendo ofertas de aquellos agentes que podrían realizar alguna de las tareas.

Un agente que recibe el anuncio de una tarea responde con una oferta por esa tarea, indicando cuan bien él considera que puede realizarla. El administrador recoge las ofertas de los contratistas potenciales y adjudica la tarea al mejor postor. Cada agente adjudicatario realiza su tarea y envía los resultados al administrador quien estará encargado de recibir y sintetizar todos los resultados. En la decisión del mejor postor, se debe tomar en cuenta los mecanismos para poder establecer métricas sobre el desempeño de cada agente.

Durante la realización de una tarea, un contratista puede convertirse en administrador y subcontratar los servicios de otros agentes. Si bien el protocolo de redes de contratos fue considerado originalmente como una técnica de negociación, es realmente un mecanismo de coordinación para la asignación de tareas. Entre sus ventajas se pueden resaltar la asignación dinámica de tareas y el balance de carga implícito, los agentes ocupados no necesitan responder a un anuncio de tarea. Algunas de sus limitaciones son la falta de detección o resolución de conflictos, la imposibilidad de interrumpir el proceso para atender tareas críticas y el hecho de hacer un uso muy intensivo de la comunicación.

3.2.2 Mecanismos de negociación

Una forma frecuente de interacción que ocurre entre agentes con diferentes objetivos es denominado negociación. El término *negociación* ha sido empleado con diferentes significados en la literatura relacionada a los sistemas multi agente. La palabra negociación ha sido usada como para referirse a distintos procesos de coordinación y resolución de conflictos en sistemas multi agente generales. Estos procesos, podían consistir en la resolución donde se tienen diferentes objetivos en planificación, resolución de restricciones por el acceso a recursos, comunicación de cambios de plan, etc. La negociación entre agentes estará caracterizada por:

- La existencia de alguna forma de conflicto que debe ser resuelta de manera descentralizada por agentes con intereses personales.
- Estos agentes actuarán bajo las condiciones de racionalidad acotada e información incompleta.
- El resultado social de la negociación será un trato, generado mediante la comunicación de los agentes, quienes usualmente intercambiarán propuestas y contrapropuestas en base a algún protocolo de negociación.

En un contexto de negociación, el principal problema que se debe enfrentar es el de determinar el trato específico al cual deberían llegar los agentes. En general, existe consenso sobre cuál sería el conjunto de posibles tratos que sería razonable que se consideraran en el proceso de negociación. Este conjunto es denominado el conjunto *bargaining* o conjunto de negociación. El conjunto *bargaining* es el conjunto de todos los tratos que son racionales individualmente.

3.2.3 Mecanismos de votación

Los mecanismos de votación se encuentran entre los métodos más utilizados para resolver problemas de la vida diaria en los que un conjunto de agentes deben tomar una decisión social consensuada.

La familiaridad con este tipo de mecanismos ha llevado a que distintos investigadores propongan esta forma de elección social como un medio para lograr que agentes artificiales interactúen de manera coordinada en un sistema multi agente.

Sin embargo, el uso de mecanismos de votación no está restringido a escenarios con múltiples agentes, las posibles acciones del agente son las alternativas sujetas a votación.

En un escenario de votación, el conjunto de resultados sociales posibles O está representado por un conjunto C de alternativas candidatas. El conjunto A de agentes participantes será referenciado como el conjunto E de electores. Cada agente $i \in E$ tiene una relación de preferencia definida sobre el conjunto C . El objetivo de un mecanismo de votación consiste en determinar una regla de elección social que tome como entrada las preferencias individuales de los agentes y produzca como salida una relación de preferencia social con algunas propiedades deseables. Straffin, listó diversos criterios de deseabilidad para los resultados obtenidos con distintas reglas de votación:

- Criterio Pareto: Si cada votante prefiere una alternativa x a una alternativa y , una regla de votación no debería producir y como ganadora.
- Criterio del ganador Condorcet: Si una alternativa x obtendría una mayoría de votos en contienda de a pares contra cada una de las otras alternativas, la alternativa x debería ser ganadora.
- Criterio del perdedor Condorcet: Una alternativa y que perdería en contienda de a pares por mayoría contra cada una de las otras alternativas, no debería ser elegida como la ganadora.
- Criterio de monotonicidad: Si una alternativa x es ganadora bajo una regla de votación, y uno o más votantes cambian sus preferencias en una manera favorable a x (sin cambiar el orden en que ellos prefieren las otras alternativas), entonces x aún debería ser la ganadora.
- Criterio de la mayoría: Si una mayoría de votantes tienen una alternativa x como su primera elección, una regla de votación debería elegir x como ganadora.

3.2.4 Mecanismos de subastas

Los mecanismos de subastas han sido utilizados en muchas aplicaciones que involucran la asignación eficiente de bienes, tareas y recursos. Un mecanismo de subastas clásico está compuesto por un conjunto de agentes A donde uno de los agentes cumple el rol de subastador y los agentes restantes son los oferentes.

Los mecanismos de subastas tienen algunas diferencias significativas con respecto a los mecanismos de votación. Con votaciones, el resultado social compromete a todos los agentes, mientras que en las subastas el resultado social es un trato entre sólo dos agentes: el subastador y uno de los oferentes. Por otra parte los protocolos de votación buscan mejorar el beneficio social, mientras que en las subastas el subastador busca maximizar su propio beneficio.

El escenario de subastas clásico asume que el subastador desea vender al precio más alto posible, mientras que los oferentes desean comprarlo al precio más bajo posible. Los mecanismos de subastas han utilizado distintos protocolos de subastas que varían de acuerdo a alguna de las siguientes dimensiones:

- Determinación del ganador:
 - Primer precio: Gana el que ofreció más.
 - Segundo precio: Igual que el anterior pero sólo paga la cantidad correspondiente a la segunda oferta más alta.
- Visibilidad de las ofertas:
 - A viva voz: Cada agente puede ver la oferta que realizan los otros agentes y las ofertas son conocimiento común.
 - A sobre cerrado: Los agentes no son capaces de determinar las ofertas realizadas por los otros agentes.
- Presentación de ofertas:
 - Ronda única: Existe una sola ronda de ofertas, después de la cual el subastador asigna el bien al ganador.

- Ascendente: El precio comienza bajo y las ofertas se realizan en cantidades incrementales.
- Descendente: Se comienza con un valor alto, y el precio se decrementa en las rondas sucesivas.
- Valor del ítem subastado:
 - Subastas de valor privado: el valor del bien depende solamente de las propias preferencias del agente. El oferente ganador no revenderá el bien ni obtendrá una utilidad de hacer alarde enfrente de los otros de ninguna manera, el valor dependería de las valoraciones de los otros agentes. Ejemplo: Subastar una pieza de pan que el ganador comerá.
 - Subastas de valor común: el valor que un agente da a un ítem depende completamente del valor que los otros agentes dan al mismo, los cuales son iguales a los del agente por simetría. Ejemplo: el valor de los bonos de tesoro depende completamente de sus posibilidades de reventa.
 - Subastas de valor correlacionado: El valor que asigna un agente depende parcialmente de sus propias preferencias y parcialmente de las valoraciones de los otros agentes. Ejemplo: un agente compra una pintura que le gusta pero mantiene abierta la posibilidad de volver a venderla en el futuro.

3.2.5 Resolución de conflictos por arbitraje

Los mecanismos de interacción presentados previamente son participativos, en el sentido de que todos los agentes pueden influenciar o aportar al resultado social. Las interacciones en los sistemas multi agente no siempre son tan democráticas. En algunas situaciones, la resolución de conflictos es delegada en una entidad o agente especial denominado árbitro. El árbitro utiliza algún criterio predeterminado para resolver los posibles conflictos entre agentes.

Los sistemas de arbitraje en Sistemas Multi-agente pueden ser utilizados por ejemplo para resolver los conflictos generados por el acceso a un recurso compartido. En este caso, el agente árbitro recibe los requerimientos para acceder al recurso desde los agentes que lo necesitan. El recurso es asignado por el árbitro a alguno de los agentes que lo requieren en base a algún criterio particular. El criterio para asignar el recurso puede ser tan elemental como el de utilizar el orden de arribo de los requerimientos. También es posible que el árbitro utilice un sistema de prioridades, donde cada cliente potencial del recurso, tendrá asignada una prioridad de acceso. Los requerimientos de agentes con más alta prioridad podrán relegar los requerimientos de agentes menos prioritarios.

4 ESTRUCTURA DE LA PROPUESTA DE LA ARQUITECTURA

En este capítulo se tratan los aspectos importantes en el desarrollo de la arquitectura, tales como los elementos que la conforman, la naturaleza del sistema, las estrategias de comunicación y coordinación que se utilizaron, así también como los agentes robóticos empleados y los mecanismos para realizar la programación.

De igual manera se hace una referencia especial a la herramienta utilizada denominada ROS, la cual facilitó gran parte del desarrollo, lo que permitió centrarse más en llevar a cabo la arquitectura sin preocuparse por programar rutinas a bajo nivel para los agentes robóticos.

4.1 NATURALEZA DEL SISTEMA

La arquitectura aquí presente se desarrolla en un esquema partiendo del tipo de robots y las características que se desean tener para ver qué tipo de tareas se pueden resolver, por lo que dicha arquitectura no debe ser ligada a un tipo de tarea específica, o a sólo un tipo de robot, por lo que podemos decir que nuestra arquitectura contiene elementos heterogéneos.

Principalmente se consideran dos tipos de agentes robóticos con características similares:

- Agentes móviles terrestres
- Agentes móviles aéreos no tripulados

Una de las tareas que se tenía en mente para realizar la propuesta es la exploración de entornos desconocidos, por lo que se decidió que el tipo de sistema debe ser de tipo **cooperativo**, por lo que los agentes interactúan de tal forma que juntos logran una misma tarea en común.

En cuanto al comportamiento de los agentes, se decidieron los siguientes puntos:

- Todos los agentes robóticos son de tipo reactivo/deliberativo
- Cada agente posee un conjunto de sensores propios
- La movilidad de los agentes es de tipo diferencial (modelo simplificado)

Específicamente, se tomaron en cuenta dos agentes con los que se cuentan físicamente en el laboratorio MOVIS, para poder realizar las pruebas físicas a la arquitectura.

- Robot móvil terrestre Pioneer P3-DX
- Robot aéreo no tripulado (Dron) Parrot Bebop 2

A lo largo de este capítulo se describirán los elementos que conforman la arquitectura, tanto los agentes robóticos, como la planeación en comunicación y las estrategias de coordinación y asignación de tareas que se realizaron.

4.1.1 Descripción de los elementos robóticos

4.1.1.1 Pioneer P3-DX

Este robot móvil de la empresa Adept MobileRobots es un robot compacto de manejo diferencial que permite realizar tareas de exploración y navegación, a través de una interfaz programable utilizando librerías escritas para C++ y Matlab, aunque existen también librerías independientes como ROSAria que permite el uso de este móvil con el sistema operativo ROS, el cual fue el sistema operativo empleado para hacer funcionar la arquitectura.

Cuenta con ocho sensores ultrasónicos de base que permiten conocer distancias frontales de obstáculos, aunque no son de una precisión completamente acertada y con tres ranuras para baterías.



Figura 4.1. Robot móvil P3-DX de Adept MobileRobots

Es posible manipular el robot a través del movimiento independiente de cada llanta, debido a que no están sujetas a un mismo eje, y cada una cuenta con un motor por separado. De acuerdo a la coordinación de las llantas, es la forma en la que el robot va a desplazarse, según el modelo que utilicemos para describir sus movimientos.

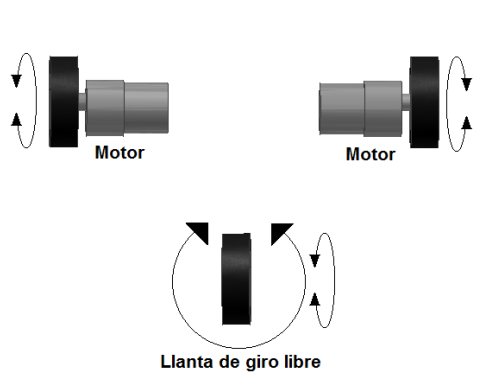


Figura 4.2. Configuración típica de un robot móvil diferencial

La gran facilidad de este tipo de robots móviles es precisamente el control diferencial, ya que este tipo de control permite manejar vueltas sobre su eje y cambiar de dirección sin necesidad de moverse para lograr un nuevo ángulo de inclinación, los movimientos de las llantas de este tipo de robots se aprecian en la Figura 4.3.

Los movimientos en un esquema diferencial se logran a través de aplicar la misma velocidad a las llantas, diferenciando únicamente cuándo se debe aplicar el movimiento hacia adelante o hacia atrás en las llantas, dado que los motores son capaces de reaccionar en reversa.

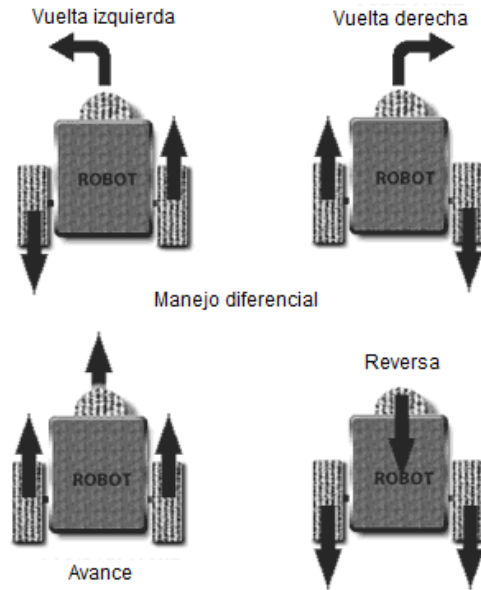


Figura 4.3. Movimiento de las llantas en un modelo diferencial

4.1.1.2 Dron Parrot Bebop 2

Este dron es fabricado por la empresa Parrot, es un dron semi profesional que está diseñado para ambientes grandes cerrados y para exteriores. Tiene sistemas de estabilidad muy buenos que permiten que el dron mantenga su posición a pesar de existir ligeras corrientes de aire, además de tener la ayuda de un SDK para poder reportar algunos datos básicos como su posición espacial o implementar la función de retorno a casa, ingresando coordenadas geográficas para que ubique su hogar.



Figura 4.4. Robot aéreo no tripulado Parrot Bebop 2

Algo importante que se puede destacar es la falta de adición de nuevos elementos, por lo que nos vimos en la necesidad de limitar las tareas posibles con este dispositivo, además de necesitar comunicación directa Wifi con una computadora para poder ejercer las funciones del SDK provisto por la empresa.

Es posible llevar un manejo diferencial en este tipo de drones, ya que son capaces de imitar los movimientos propios de este modelo, a pesar de poder realizar más movimientos posibles, lo cual complica el modelo a un modelo en 3 dimensiones. Por simplicidad y aplicación de la arquitectura, se decidió utilizar el modelo diferencial. Para lograr este modelo en un dron, se debe tomar en cuenta el número de hélices con el que cuenta. Una vez hecho esto, se impulsa al dron con un par de ellas, de manera que gire sobre su eje en vez de avanzar. Aunque este tipo de movimientos son más lentos, permiten tener una certeza más asegurada en la localización espacial, y por ende, una facilidad en la ubicación espacial en los algoritmos.

Una de las problemáticas físicas de este tipo de robots es el funcionamiento con el que se mantiene en el aire, ya que debido a la potencia con la que las hélices giran para mantener a un robot de este tamaño en el aire, se genera una corriente de aire que afecta la precisión de la posición en espacios demasiado pequeños y cerrados. Esto también afecta la capacidad del robot para pilotear y particularmente puede resultar peligroso hacerlo cerca de alguna persona.

4.2 MODELO DE COMUNICACIÓN

4.2.1 Especificaciones físicas

Una vez que se tomaron en cuenta las capacidades individuales de cada agente, lo que se hace es adaptarlos a un modelo de comunicación. Para la arquitectura, se decidió realizar un modelo centralizado en la comunicación, debido a que muchos elementos trabajan a través de redes inalámbricas wifi LAN (802.11). A la parte central se le conoce como Access point (AP).

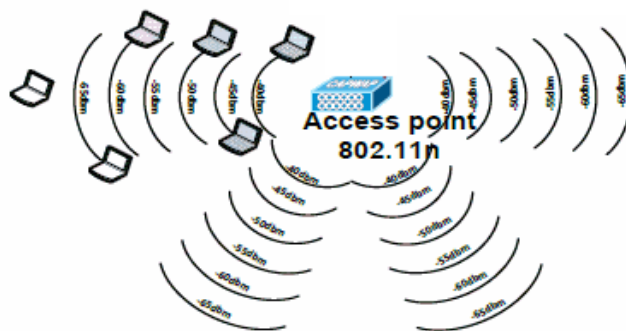


Figura 4.5. Modelo de LAN para la arquitectura

Este tipo de comunicaciones requiere que el AP esté colocado de manera central, ya que las señales irradian en forma circular con respecto al punto de origen. Generalmente el alcance

de una red Wifi es de 5 a 10 metros, dependiendo del estándar y las especificaciones técnicas de las antenas en los dispositivos.

Específicamente, para el funcionamiento de nuestra arquitectura, el AP es una computadora que engloba todos los procesos de ruteo a los diferentes dispositivos, lo que nos permite tener una mejor gestión sobre la información que se está transmitiendo.

4.2.2 Intercambio de mensajes

Ya se había mencionado anteriormente el concepto de ‘mensaje’ en una comunicación de sistemas multi robot. En esta sección vamos a especificar el protocolo de mensajes que se utilizó.

Existen tres grupos de mensajes, los cuales categorizan dichos mensajes de acuerdo a la función que cumplan:

- **Organización del sistema.** Son mensajes que permiten llevar el control de los eventos en el sistema, como la adición o reincorporación de un elemento robótico (mensajes de saludo), mensajes de progresión de tareas, entre otros tipos de mensajes.
- **Información personal.** Son mensajes que describen características de un agente robótico, tales como la posición espacial (punto dentro del plano) en la que se encuentra, estado operativo del robot, o datos e información arrojados por los sensores que el robot posea.
- **Llamadas a procedimiento remoto.** Son mensajes que permiten solicitar alguna acción predeterminada a un robot en específico.

Es necesario en una arquitectura especificar todos los tipos de mensajes que los robots intercambian, ya que esto permite identificar el contenido de los mismos, además de servir para saber cómo se va a procesar. Los mensajes nos permiten entablar una comunicación, pero para una mejor organización del sistema, es necesario que existan diferentes tipos de mensajes con su propia estructura, con el fin de saber qué se está solicitando o comunicando y cómo proceder ante eso.

En forma técnica, los mensajes son estructuras de datos que contienen campos, cada uno cumpliendo la función de describir el mensaje, o llevar el contenido bruto del mismo. Generalmente cuando mandamos un mensaje, es necesario incluir de dónde proviene (emisor), y a quién o dónde va dirigido (receptor). Los demás campos genéricos pueden incluir metadatos del mensaje como la fecha y hora, o alguna circunstancia ajena al contenido central. La raíz del mensaje, es decir el contenido puro, puede establecerse en algún tipo de dato específico si sabemos que, por ejemplo, el mensaje central va a comunicar una cifra numérica o simplemente un valor binario.

Los mensajes son muy específicos tanto a los protocolos y métodos de comunicación empleados en la arquitectura, como al objetivo general que se esté buscando cubrir, por lo que no siempre podremos encontrar un marco de referencia común completo para todos los sistemas. En la Tabla 4.1 se muestran los tipos de mensajes utilizados en la arquitectura.

Tipo de mensaje	Grupo	Descripción
Saludo	Organización del sistema	Mensaje que envía un robot al sistema central de comunicación para reportar su entrada.
Keep-alive	Organización del sistema	Mensaje que permite dar a conocer que aún se está conectado a la central.
Desconexión	Organización del sistema	Mensaje utilizado para informar a la central el abandono de un robot del sistema.
Asignación de tareas	Organización del sistema	Mensaje genérico que permite dar a conocer a un robot lo que debe realizar.
Finalización de tarea	Organización del sistema	Mensaje que le informa a la central que una tarea ha sido completada.
Comando de movimiento	Organización del sistema	Mensaje que indica el movimiento que debe ejercer un robot (velocidades lineales y angulares).
Mensaje de confirmación	Organización del sistema	Mensaje que sirve para responder si se ha podido atender o no una petición remota. Útil cuando los procedimientos remotos no deben retornar información.
Posición espacial	Información personal	Mensaje que permite dar a conocer el punto en donde se encuentra un robot (en ejes tridimensionales x,y,z).
Información sensorial	Información personal	Mensaje que contiene datos extraídos de los sensores de un robot (principalmente ultrasónicos, láser o imagen de cámaras).
Estado operativo	Información personal	Mensaje utilizado para dar a conocer el estado general del robot.
Petición de información remota	Llamada a procedimiento remoto	Mensaje que realiza una petición a un robot acerca de información personal, generalmente se espera una respuesta de este tipo.
Petición de procedimiento remoto	Llamada a procedimiento remoto	Mensaje que indica que el robot necesita realizar alguna acción pre programada dentro de él.

Tabla 4.1. Lista de mensajes en el protocolo

4.2.3 Organización de la comunicación

Se decide que la arquitectura puede implementar un rango de comunicación ilimitada, para la cual se proponen las estrategias de comunicación de pizarra, envío de mensajes y política cooperativa. La estrategia de pizarra se propone como un método centralizado para resolución de conflictos, el envío de mensajes y la política cooperativa se proponen como métodos descentralizados para la resolución de conflictos individuales.

Es importante señalar que en las estrategias de coordinación basadas en comunicación con rango ilimitado la información se difunde a todos los robots en el ambiente y no sólo a los que se encuentren cerca como en la comunicación con rango limitado, es decir que ahora el

grupo de agentes robóticos estará formado por todos los robots que se encuentren en el ambiente.

En cuanto a la comunicación directa, sólo se reserva en casos específicos de coordinación necesarios para el sistema, por lo que el método preferido es la pizarra.

4.2.3.1 Sistema de pizarra empleado

Para la implementación de este método se considera a cada robot como una estructura independiente de los demás y en la que se guarda la información particular de cada robot.

Recordamos que la pizarra como tal es una estructura de datos en la que, cada robot publica en un espacio dentro de la pizarra su ID, el árbol que cada robot va construyendo asociado a su región segura y su posición actual, entre otros mensajes. Así, en las figuras y tablas en esta sección se muestra la forma en que trabaja esta estrategia de comunicación.

La primera parte es la estructura de la pizarra. Se tomó la decisión de utilizar un esquema de pizarra dividida en tópicos, para poder consultar la información de manera ordenada. El número de tópicos queda definido individualmente para cada tarea, de acuerdo a las necesidades de la misma. Un ejemplo de tópico es uno consistiendo de información arrojada por los sensores de los robots, por lo que los agentes que deseen conocer la información del entorno deben consultar los tópicos que en ella se vayan depositando. Cabe aclarar que la información que se publica en cada tópico dentro de la pizarra puede no ser del mismo tipo, por lo que los mensajes tienen explícitamente un tipo de acuerdo a lo descrito en la sección 4.2.2.

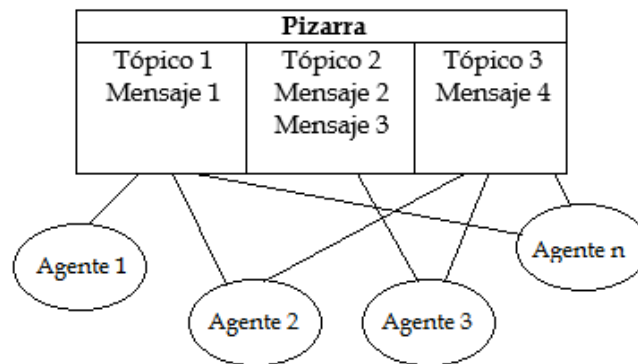


Figura 4.6. Representación de la pizarra por tópicos

En la Figura 4.6 se aprecia el funcionamiento de los agentes en la pizarra. Cada agente es capaz de ‘suscribirse’ a un tópico dentro de la pizarra, con el fin de leer los contenidos que son publicados en esta sección. Un agente también puede publicar dentro de estos tópicos, de acuerdo a los permisos que tenga cada uno de ellos para hacerlo. Las relaciones que los agentes pueden tener con los tópicos son de tipo lectura, publicación, o ambos.

La Figura 4.7 está dividida en tres pequeños cuadros los cuales muestran el proceso de comunicación que lleva a cabo un robot para poder coordinar sus movimientos sobre el ambiente desconocido que está explorando. En el primer cuadro se muestra el estado inicial

de ésta estrategia de comunicación, se tienen los robots del ambiente y el tópicos de la pizarra donde cada robot guarda, actualiza y consulta información.

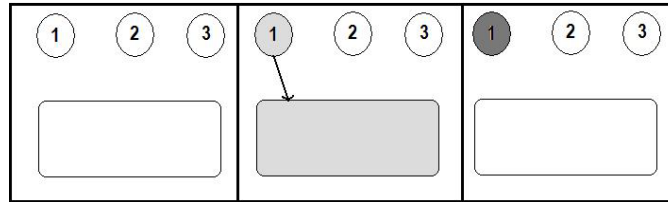


Figura 4.7. Proceso de la estrategia de coordinación pizarra para el primer robot

En el segundo cuadro de la misma Figura 4.7 se muestra el paso dos del proceso, el cual consiste en elegir al robot de menor ID en el ambiente y que no haya actualizado información en la pizarra; en este caso se elige al robot 1, el cual actualiza su información en la sección de la pizarra que le corresponde, cuando finaliza el robot corta la comunicación con la pizarra y es marcado con una bandera para indicar que él ha actualizado su información y con ello evitar ser reconsiderado para actualizar información en la pizarra antes de que todos lo hagan.

Cabe mencionar que el momento en que el robot 1 y en general cualquier robot accede a un tópicos de la pizarra, éste se pone en estado OCUPADO (cuadro gris). Una vez que termina el proceso de actualización de información, el estado del tópicos se vuelve a DESOCUPADO (cuadro blanco). Finalmente, el robot 1 se elimina del grupo que desea actualizar información sobre este último. Esto se realiza para gestionar de manera adecuada cada tópicos, ya que cada uno de ellos actúa como una región crítica donde muchos agentes pueden ver y publicar mensajes.

El proceso se repite con cada uno de los robots del ambiente, hasta que todos terminen de publicar sus datos, en la Figura 4.8 se muestra el proceso que sigue el robot 2 para actualizar su información en el tópicos de la pizarra y en la Figura 4.9 se muestra el proceso que sigue el robot 3.

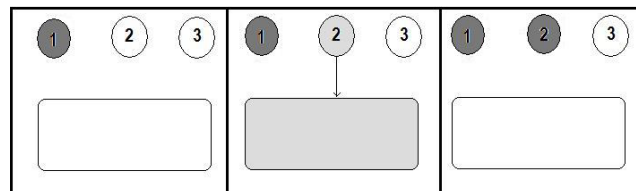


Figura 4.8. Proceso de la estrategia de coordinación pizarra para el segundo robot

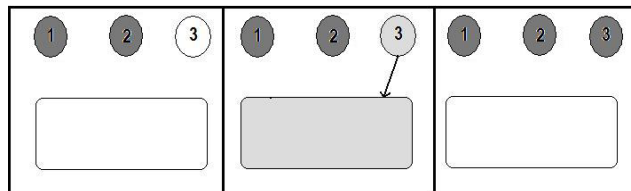


Figura 4.9. Proceso de la estrategia de coordinación pizarra para el tercer robot

Una desventaja de esta estrategia de coordinación es el tiempo que cada robot debe esperar para poder acceder a un tópico de la pizarra con el fin de actualizar sus datos y conocer la información de otros robots.

4.3 MODELO DE COORDINACIÓN

En el modelo de coordinación, se estudiaron algunas técnicas para colocar la que mejores resultados tiene en la implementación, y de acuerdo a la tarea específica a resolver, además de las características tanto físicas como sociales de cada agente robótico.

4.3.1 Comportamiento y naturaleza de los agentes robóticos

Lo primero que se debe discutir es el comportamiento que van a adoptar los robots dentro del sistema.

De acuerdo a las características, existen dos tipos de agentes móviles:

- Agente robótico terrestre
- Agente robótico aéreo

El comportamiento de ambos agentes es de tipo **cooperativo**. También, los agentes planifican sus tareas de una forma reactiva/deliberativa, lo que quiere decir que en algún momento adoptan ambas formas, pero la forma predominante es la reactiva. Esta naturaleza nos lleva a tener un sistema cuya coordinación principal es la **especialización**. Si recordamos esta técnica, los agentes se especializan en resolver cierto tipo de tareas de acuerdo a las características propias de los robots y al entorno que los rodea.

En la arquitectura, el agente que se agrega exclusivamente para la coordinación de la comunicación también ofrece tareas para que los robots las exploren, por lo que nos encontramos ante una arquitectura **centralizada**. La técnica de coordinación utilizada por este agente central (no robótico) se conoce como **coordinación de acciones**. Específicamente se utilizará la **coordinación por planificación**, donde un agente central planifica las tareas que otros agentes deben realizar, con el fin de evitar colisiones y obtener resultados de manera clara y precisa.

Se prefieren modelos centralizados a los distribuidos, ya que esto permite desarrollar sistemas más simples con componentes de bajo costo, ya que no es necesario que cada agente posea un procesador potente para poder gestionar sus propias acciones, el dominio de su entorno, y la interacción con otros agentes.

Por parte de la asignación de roles, los únicos roles que tenemos por tipo de agente son tres:

- Agente terrestre
- Agente aéreo
- Coordinador

Los agentes terrestres tienen la capacidad de mapear su entorno en dos dimensiones a través de un sensor láser de barrido de 270°, además de contar con una corona de 8 sensores ultrasónicos en la parte frontal. Mientras tanto, los agentes aéreos tienen limitantes más

importantes en cuanto a sensores, ya que sólo pueden medir la altura con respecto del suelo a través de un sensor ultrasónico equipado en la parte inferior de los mismos. En cuanto al rol de coordinador, la única especificación de este tipo de agente es que se trata de una computadora personal con comunicación inalámbrica, capaz de abarcar comunicación a todo el entorno de trabajo, donde todos los procesos importantes se están ejecutando para la coordinación de los robots.

4.3.2 Especialización y coordinación de acciones por planificación

Los agentes robóticos adquieren la coordinación por especialización principalmente debido al comportamiento primariamente reactivo, puesto que reaccionan más a los obstáculos y otros agentes que tengan frente a ellos, más que planificar sus acciones de manera premeditada.

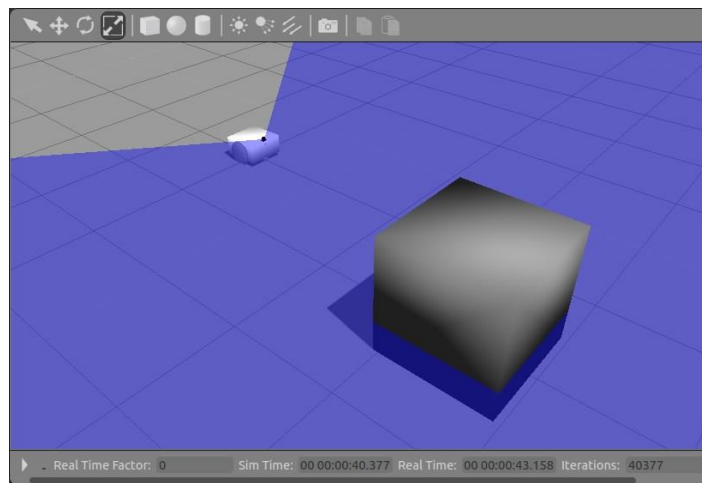


Figura 4.10. Simulación demostrando el rango láser de los agentes terrestres en Gazebo

La Figura 4.10 muestra los objetos que puede detectar cada agente terrestre, dentro del rango visible (azul) del sensor láser que poseen. La primera acción de los robots es la evasión de obstáculos, mientras que esto resulta fácil con los obstáculos fijos, es más difícil cuando se tienen dos o más agentes terrestres en rangos próximos. Es un buen caso de estudio que generalmente los comportamientos de evasión en ambos robots van a ser activados, por lo que vale la pena observar las reacciones que cada uno de ellos va a tener con respecto al otro, sobre todo si la programación de las rutinas primitivas de evasión es la misma en ambos.

Con respecto a los agentes aéreos, la especialización se presenta debido a la falta de sensores de proximidad frontales/laterales, únicamente cuentan con una cámara de visión de ojo de pez, la cual no facilita la tarea, por lo que se decide que la forma de evitar colisiones es a través de dos maneras:

- **A través de las posiciones**, ya que permite que los agentes aéreos naveguen en zonas donde otros agentes no estén próximos.
- **A través de niveles**, en el que cada agente vuela a una altura prudente, diferente a la de otros agentes, ya que el modelo de los agentes aéreos está simplificado a un

modelo diferencial, únicamente se moverán en un plano bidimensional sin posibilidad de estorbarse.

La especialización únicamente funciona cuando los agentes se encuentran navegando en su entorno, por lo que cada uno de ellos ya tiene presente la tarea que debe realizar. En realidad los agentes tienen la función a cumplir pre programada dentro de sus comportamientos, sin embargo esta puede cambiar según los intereses del sistema y los objetivos generales.

Con respecto a la coordinación de acciones por planificación, los agentes reportan los descubrimientos que van realizando a través de sus terminales, mientras que el agente central se encarga de administrar los tópicos de la pizarra y otorga los permisos de escritura a los agentes en cada uno de ellos. Cuando se junta suficiente información en un tópico dedicado, el agente central se da a la tarea de estructurar esta información, de tal manera que se publique en forma de un mensaje legible por todos los agentes que así lo requieran, además de poder ser presentada gráficamente para los usuarios que lo soliciten.

4.3.3 Política cooperativa para resolución de conflictos

Lucia Pallottino, Vincenzo Scordio, Antonio Bicchi y Emilio Frazzoli presentan en [32] una idea novedosa la cual consiste de varias reglas las cuales son englobadas en las palabras política cooperativa. Así en lo siguiente nos referiremos de ese modo a su trabajo, el cuál presenta una política cooperativa para direccionar múltiples vehículos entre configuraciones iniciales a configuraciones finales asegurando que sus caminos se encuentren libres de colisiones.

Esta política se basa en que todos los agentes son cooperativos y siguen las mismas reglas de tráfico, estas reglas son aplicadas de acuerdo a la información local disponible, el objetivo de este trabajo está enfocado en el problema de evitar colisiones entre vehículos en movimiento en un ambiente compartido además de poder permitir la integración de nuevos agentes a la tarea.

Para este trabajo se tomaron las ideas expuestas en el artículo [22] y se han adaptado a la arquitectura, conservando la idea de que cada robot es independiente de los demás y que en cada uno de ellos se guarda información propia de cada robot y el envío de un mensaje con información consiste en tomar el robot de menor ID, cada robot del grupo escribe su información en algún tópico de la pizarra.

4.3.3.1 Reglas para la política cooperativa

Para explicar las reglas en las que se basa ésta estrategia de coordinación es necesario definir la notación utilizada para referirnos a la posición que ocupa en determinado momento un robot. Se le llama configuración a la posición en la que se encuentra en un momento dado un robot, así, una configuración está compuesta por un punto (x, y) donde, x y y son números enteros. Denominando a la posición como g .

Se le llama plan a un par de posiciones $(g_{ini}; g_f)$, las cuales son una configuración inicial (x, y) y una configuración final (x, y) respectivamente. De ésta manera definimos plan $P = (g_{ini}; g_f)$ como un movimiento seguro de un robot sobre el ambiente, donde, g_{ini} es la

posición actual del robot y g_f es la posición candidata. Durante el tiempo de exploración cada robot va coleccionando todas sus posiciones iniciales g_{ini} en un conjunto llamado G_i y sus posiciones finales g_f en un conjunto llamado G_{if} , de ésta manera el conjunto G_i es definido como:

$$G_i = \{g_{ini}, ini = 1, \dots, n\}$$

La cual denota el conjunto de configuraciones iniciales de los robots en un determinado tiempo donde i indica el ID del robot asignado por el agente central.

El conjunto G_{if} es definido como:

$$G_{if} = \{g_f, f = 1, \dots, n\}$$

La cual denota el conjunto de configuraciones candidatas de los robots en un determinado tiempo donde i indica el ID del robot asignado por el agente central.

Las reglas bajo las cuales la política cooperativa es aplicable son admisibilidad, posicionamiento seguro, seguridad y liveness, estas reglas se explican detalladamente a continuación.

4.3.3.1.1 Admisibilidad

Considerando un ambiente en el cual nuevos robots pueden preguntar a otros robots, si pueden entrar a interactuar en el escenario en un tiempo dado y con un plan arbitrario donde, el plan consiste de una posición inicial y una posición candidata. En este caso, es importante tener condiciones para decidir eficientemente si se acepta o se rechaza la nueva petición.

La nueva propuesta debe cumplir con condiciones las condiciones $P1$ y $P2$ para asegurar que el plan con el que entran a interactuar en el ambiente esté libre de colisiones dadas las siguientes propiedades:

$P1$: Un conjunto de configuraciones $G_i = \{g_{ini}, ini = 1, \dots, n\}$ es considerado inseguro, si existe un conjunto de configuraciones $G_{if} = \{g_f, f = 1, \dots, n\}$ tal que conduzcan a alguna colisión.

$P2$: Un conjunto de configuraciones $G_{if} = \{g_f, f = 1, \dots, n\}$ será bloqueado o rechazado, si existe un conjunto de configuraciones iniciales $G_i = \{g_{ini}, ini = 1, \dots, n\}$ tal que conduzcan a un bloqueo.

Un robot es aceptado por su plan $P = (g_{ini}; g_f)$ si verifica el predicado $\neg P1(G_i) \wedge \neg P2(G_f)$, es decir, que no hay colisiones con los planes de los robots que se encuentran ya operando en el ambiente.

4.3.3.1.2 Posicionamiento seguro

El primer paso de esta política de coordinación es verificar que cada agente esté bien posicionado y que guíen a un sistema dinámico bien posicionado, esto se refiere a que, desde el inicio y durante el proceso de exploración del ambiente cada robot debe cuidar no invadir las regiones seguras de otros robots.

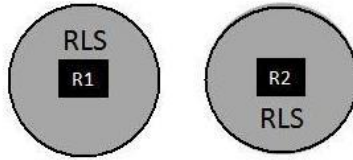


Figura 4.11. Configuración correcta para las regiones seguras de los robots.

En la Figura 4.11 se muestra como las áreas seguras de los robots no deben traslaparse en un inicio y durante el proceso.

Esta regla es definida por el teorema 1, cuya demostración se puede consultar en el trabajo [32].

Teorema 1: *El sistema es bien posicionado, si para toda condición inicial de los agentes el interior de sus regiones seguras RS son disjuntas.*

4.3.3.1.3 Seguridad

Ésta regla de la política cooperativa, propone que las regiones seguras de los robots no se traslapen durante el proceso de exploración, si en algún momento dado esto llegara a ocurrir, el robot no debe avanzar, en este caso se busca por una nueva posición candidata, en caso de no encontrar el robot permanecerá en su posición, la Figura 4.12 muestra un traslape de regiones seguras de dos robots, este problema es el que se corrige con la regla de seguridad.

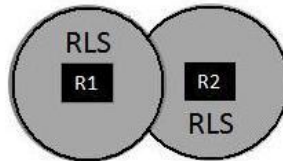


Figura 4.12. Configuración incorrecta con regiones seguras traslapadas en los robots

Como podemos darnos cuenta la descripción que se acaba de hacer sobre la regla de seguridad tiene que ver con la propiedad P1, de la que se deriva el teorema 2 cuya demostración se puede consultar en el trabajo [32].

Teorema 2: *Si las regiones seguras RS de al menos dos robots se traslapan, la propiedad P1 es verificada. En este caso una política de retroceso o re planificación es implementada. Dado que los agentes siempre se encuentran dentro de su región segura local RSL, se puede asegurar que el sistema se encuentra libre de colisiones.*

4.3.3.1.4 Liveness

Liveness o vivacidad tiene que ver con la propiedad P2, y se basa en la definición de una condición que mantiene separadas a las regiones seguras, asociadas a las configuraciones objetivo de los robots.

En otras palabras la vivacidad de un robot se encarga de mantener a cierta distancia los puntos objetivos de los robots para evitar traslapes de estas regiones y con ellos posibles colisiones.

Ésta regla es importante debido a que mantener una distancia considerable entre los puntos objetivos de los robots ayuda a evitar bloqueos como el que se muestra en la 39.

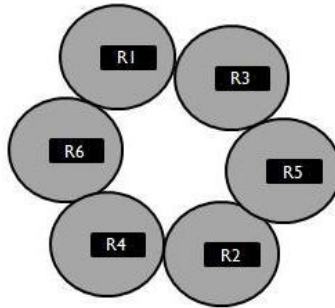


Figura 4.13. Bloqueo durante el proceso de exploración debido a regiones seguras

De la importancia de evitar bloqueos durante el proceso de exploración se define la siguiente propiedad.

P3: Un conjunto de configuraciones objetivo $G_{if} = \{g_f, f = 1, \dots, n\}$ forma un grupo si la distancia entre las configuraciones finales es menor al radio de la región segura.

De la propiedad anterior se deriva el siguiente teorema, que para su demostración se puede consultar el artículo [32].

Teorema 3: *La propiedad P2 es válida para la política de coordinación si P3 es válida.*

En otras palabras la separación de las configuraciones objetivo es una condición necesaria para evitar conducir a un bloqueo en esta política, una condición suficiente es presentada por el siguiente teorema:

Teorema 4: *Considere dos vehículos tales que el centro de las regiones reservadas en una configuración final están a una distancia mayor a $2d+4$, donde d es la distancia Euclidiana entre dos configuraciones.*

Esta política permite a los vehículos alcanzar su objetivo destino en un tiempo finito, y como condición inicial se tiene que las regiones seguras son disjuntas.

Siguiendo las reglas anteriormente expuestas se genera el siguiente algoritmo, para la coordinación de agentes robóticos exploradores de ambientes desconocidos.

4.4 CONCLUSIONES DE LA ARQUITECTURA

Las estrategias de comunicación son utilizadas en tareas como la exploración de ambientes con la finalidad de reducir el tiempo de exploración, además de evitar conflictos entre los exploradores del ambiente, en otras palabras, las estrategias de comunicación contribuyen a una correcta coordinación. Con estos mismos objetivos se presentan las estrategias de

comunicación, las cuales fueron elegidas, tomando en cuenta varias consideraciones, primero que nada sus características individuales, su fácil adaptación a las tareas que los robots llevan a cabo en el presente trabajo, además de considerar que fueran completamente diferentes entre ellas.

De esta manera poder comparar diferentes tipos de estrategias de comunicación y así, decidir qué estrategia de comunicación es de mayor utilidad para la tarea de exploración de ambientes desconocidos. Por lo que, las estrategias de comunicación elegidas y explicadas en éste capítulo son, con rango limitado, utilizando en menor medida mensajes con rango ilimitado, pizarra estructurada en tópicos, política cooperativa para la resolución de conflictos y coordinación basada en especificación, junto con coordinación de acciones por planificación central utilizando un agente central no robótico.

4.5 INSTRUMENTACIÓN DE LA ARQUITECTURA

En esta sección se describen las herramientas de software y hardware que se utilizaron para la implementación de la arquitectura en agentes robóticos, tanto simulados como reales. En un panorama general, se espera que se traten temas acerca del sistema operativo robótico llamado ROS, de simuladores como Gazebo, y de algunas librerías y conexiones para los agentes robóticos.

4.5.1 ROS

Como punto de partida en el presente trabajo, todas las partes funcionales de la arquitectura están implementadas bajo el sistema operativo ROS (Robotic Operative System por sus siglas en inglés).

ROS es un meta-sistema operativo open source para muchos tipos de robots comerciales. Provee servicios que los usuarios esperan de un sistema operativo, puesto que incluye abstracción de hardware, control de dispositivos a bajo nivel, implementación de funcionalidades usadas comúnmente, paso de mensajes entre procesos y administración de paquetes. Dentro de él podemos encontrar una colección de herramientas, librerías y convenciones que se concentran en simplificar la tarea de crear comportamientos robóticos complejos y robustos a través de una gran variedad de plataformas robóticas.

El 'grafo' de ejecución de ROS es una red *peer-to-peer* de procesos (potencialmente distribuidos entre máquinas) que están acoplados débilmente usando la infraestructura de comunicación de ROS. ROS implementa diferentes estilos de comunicación sobre servicios, transmisión asíncrona de datos sobre tópicos y almacenamiento de datos en un Servidor de Parámetros.

ROS fue construido para motivar el desarrollo colaborativo de software robótico, si un laboratorio tiene expertos en mapeo de entornos cerrados interiores, puede contribuir con una herramienta de clase mundial para producir mapas, que a su vez, otro grupo puede tener expertos en usar mapas para navegación compleja y puede reutilizar estos módulos, por lo que los trabajos no necesitan ser tan específicos para una plataforma robótica en específico.

ROS distribuye las tareas particulares en nodos, con posibilidad de comunicarse a través de un sistema de mensajes que funciona utilizando herramientas llamadas tópicos.

Las tareas se pueden programar utilizando lenguajes de programación con motores incluidos (C, Python y Lisp) o utilizar herramientas incluidas que realizan ya ciertas tareas.

Varias de dichas herramientas incluyen software para hacer simulación o lectura de sensores externos como sensores láser.

Una de las ventajas es que ya existen nodos programados para realizar tareas comunes, incluyendo el control de elementos robóticos de diversos fabricantes.

El sistema de comunicación de nodos en ROS es muy similar al esquema de la pizarra por tópicos, por lo que permite implementar este tipo de esquemas de una manera nativa y eficaz. Además de contar con librerías para el manejo de los robots disponibles en el laboratorio, por lo que se decidió utilizarlo completamente.

Otra de las ventajas incluidas es que se puede programar un sistema simulado y migrarse a un entorno físico de una manera simple, ya que sólo se sustituyen los nodos de simulación por los correspondientes nodos de control de robots físicos.

Como se puede ver, el modelo de comunicación entre procesos o nodos sigue una especie de boletines llamados Temas, en los cuales los nodos pueden publicar contenido o pueden suscribirse y recibir la información que ahí se publique. En la siguiente figura se aprecia los esquemas de comunicación entre nodos por servicio y por tópicos.

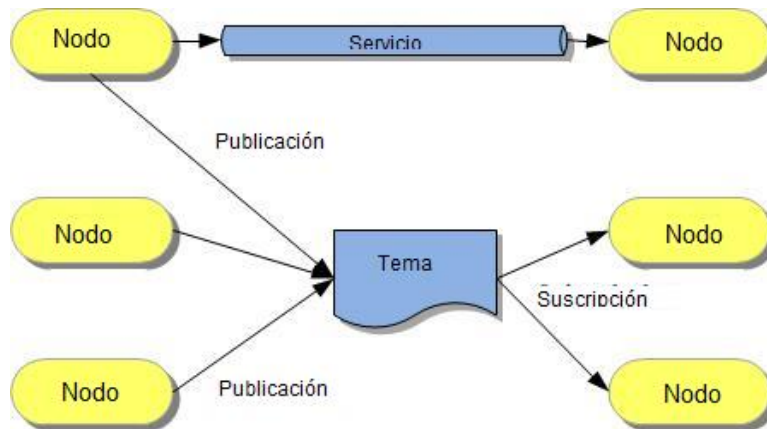


Figura 4.14. Esquemas de comunicación entre nodos de ROS

Algunos conceptos genéricos que incluye ROS:

- Manejo de paquetes y metapaquetes
- Tipos de mensajes predefinidos y personalizados
- Tipos de servicios predefinidos y personalizados
- Estructura de grafos muy similar a un grafo de comunicación que cuenta con:
 - Nodos (incluyendo un nodo maestro)
 - Servidor de parámetros

- Mensajes
- Tópicos
- Servicios
- Mochilas
- Librerías para el manejo de sensores, motores y robots comerciales a bajo nivel
- Algoritmos y técnicas de exploración programadas
- Software para facilitar la simulación (y soporte para software de simulación de terceros)

Específicamente, en el proyecto se utilizó la versión Indigo de ROS, ya que era la versión con más soporte de la comunidad, aparte de ser la versión con librerías más actualizadas y más documentación en la red.

4.5.2 Gazebo

Gazebo es un software que permite la simulación de entornos, robots, sensores, además de condiciones y motores de física, para poder implementar un algoritmo robótico antes de poderlo probar con robots reales.

Algunas de las capacidades que tiene Gazebo son modelos de robots para la mayoría de los robots comerciales, simulación dinámica fácil de utilizar, un motor de física incluido con muchas propiedades configurables, gráficos en 3D, simulación de varios tipos de sensores y cámaras (incluidas las de tipo Kinect de Microsoft), transporte TCP/IP para la comunicación entre diferentes estaciones de trabajo, entre otras más.

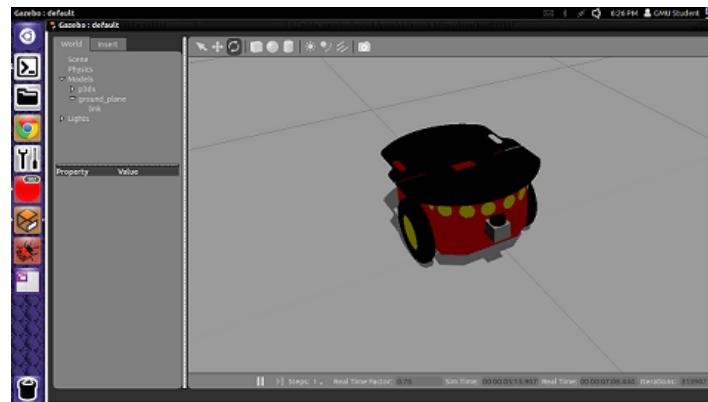


Figura 4.15. Simulación en Gazebo de un Pioneer P3-DX de AdeptMobile Robots

Dentro de Gazebo, existen dos conceptos principales:

- Mundos (Escenarios)
- Modelos (Robots)

Los mundos contienen elementos, la mayoría de ellos estáticos, que conforman el entorno de un sistema robótico, mientras que los modelos son los robots en sí, cada uno con sus propias especificaciones sobre el tipo de movimiento que pueden ejercer, los sensores que tienen ligados a ellos, además de incluir funcionalidades para ellos, entre otras especificaciones más detalladas.

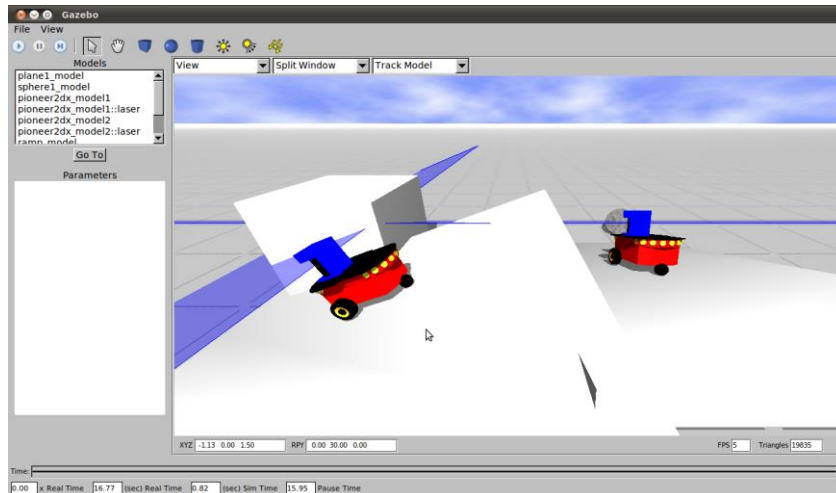


Figura 4.16. Editor de mundos y modelos en Gazebo

La manera en la que Gazebo puede interpretar un mundo o un modelo es a través de un lenguaje especial de especificación, con archivos derivados de XML con su propio esquema. Existe una alternativa gráfica para crear estos archivos de especificación utilizando una herramienta llamada *editor de mundos y modelos*. Esta herramienta permite manejar visualmente el diseño de un robot, y de su entorno, con el fin de ejecutar la herramienta de manera adecuada.

Para la comunicación de Gazebo-ROS, existe una librería que se encarga de establecer la comunicación de la herramienta simulada a comandos de ROS, por lo que la implementación de la simulación llega a ser transparente para los algoritmos robóticos que se realizan en ROS.

5 MAPEO COOPERATIVO DE ESPACIOS

En este capítulo se presenta de una manera detallada la tarea realizada para poner en marcha la arquitectura, además de reportar los resultados obtenidos con ella. De manera simulada, se trataron un par de escenarios con hasta tres agentes terrestres y uno aéreo, para ver las reacciones con respecto al entorno y las interacciones que presentaban entre ellos.

El mapeo (del inglés *mapping*) es una disciplina relacionada a la cartografía. La meta para un robot autónomo es que sea capaz de construir (o usar) un mapa o un *plan de piso* y localizarse a sí mismo en él.

Las acciones ciegas formadas por evolución pueden ser suficientes para mantener vivos a algunos animales. Para algunos insectos por ejemplo, el entorno no es interpretado como un mapa, y ellos sobreviven sólo con una respuesta inmediata. Una estrategia de navegación ligeramente más elaborada mejora dramáticamente las capacidades de los robots. Los mapas cognitivos permiten capacidades de planeación y uso de percepciones actuales, eventos memorizados y consecuencias esperadas.

5.1 FUENTES DE INFORMACIÓN EN EL MAPEO

El robot tiene dos fuentes de información: la centrada en el individuo (del inglés *idiothetic*) y la que no lo es (del inglés *allothetic*). Cuando se encuentra en movimiento, un robot puede usar métodos de navegación estimada, tales como contar el número de revoluciones por unidad de tiempo de sus llantas; esto corresponde a una fuente de tipo *idiothetic* y puede dar la posición absoluta del robot, pero está sujeta a un error acumulativo que puede crecer rápidamente.

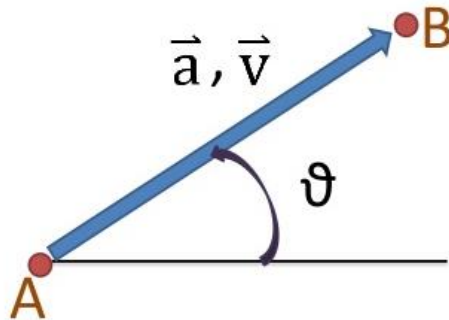


Figura 5.1. Estimación de una posición de acuerdo a una velocidad lineal y un ángulo

En la Figura 5.1 se aprecia el cálculo de una posición B, partiendo de una posición A, a la que aplicamos un par de fuerzas en vectores, dado un ángulo de inclinación. La navegación por estimación se utiliza cuando no se desean agregar otros métodos basados en sensores, tales como sensores ultrasónicos, GPS, o la colocación de algunos codificadores lineales y rotatorios, lo cual reduce el costo de producción de un robot. La utilización adecuada de este método va en el sentido en el que debe proveer una cierta cantidad conocida de electricidad o presión hidráulica a los motores del robot sobre un periodo de tiempo dado

de un punto inicial, para poder realizar sus cálculos. El error depende de los mecanismos internos del robot, de la velocidad a la que navegue, e incluso el medio por el que lo realice.

Las fuentes de tipo *allothetic* corresponden a los sensores del robot, como una cámara, micrófono, láser, sonar, etc. El problema que se presenta en esta aproximación es el *aliasing perceptual*, lo que significa que dos diferentes lugares pueden ser percibidos como el mismo. Por ejemplo, en una construcción, es casi imposible determinar la localización únicamente con mecanismos de visión (imagen), debido a que todos los corredores se ven iguales.

5.2 REPRESENTACIÓN DE LOS MAPAS

La representación interna de un mapa en un sistema puede ser métrica o topológica

- La representación métrica es la más usada por humanos y considera un espacio bidimensional generalmente, en el cual se colocan objetos. Los objetos son colocados con coordenadas precisas. Esta representación es muy útil, pero es muy sensible al ruido y es difícil calcular distancias en ella de forma precisa.



Figura 5.2. Representación métrica de un mapa a través de sensores

- La representación topológica sólo considera lugares y relaciones entre ellos. Comúnmente, las distancias entre los lugares son almacenadas. El mapa es un grafo, en el cual los nodos corresponden a lugares emblemáticos y las aristas corresponden a los caminos.

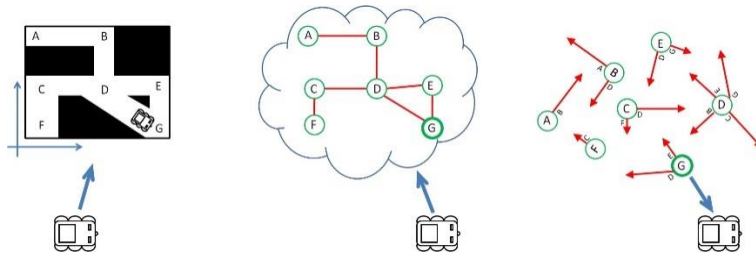


Figura 5.3. Representación topológica de un mapa con 7 lugares

Existen tres métodos principales para las representaciones de mapas: mapas de espacio libre, mapas de objetos y mapas compuestos. Todos ellos emplean la noción de rejilla, pero

permiten la afinidad de la misma para que se vuelva más o menos precisa de acuerdo a las necesidades del mapa.

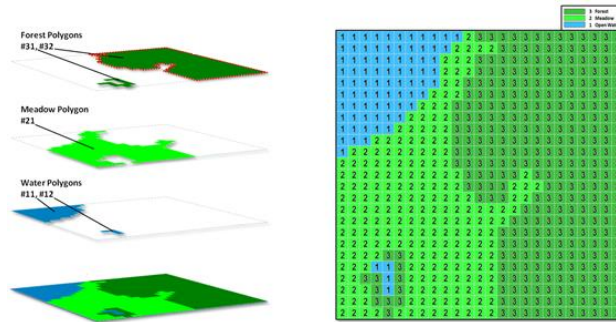


Figura 5.4. Representación por rejillas de un mapa

En el presente trabajo, se decide utilizar el mapeo por representación métrica, donde aprovechando los datos que los robots distribuyen de los sensores, pueden construir el entorno en el que se encuentran.

5.3 NAVEGACIÓN Y MAPEO A TRAVÉS DE SENSORES DE RANGO

Cuando se está navegando a través de un robot móvil con un láser equipado, éste puede realizar escaneos completos del área de trabajo que maneje cada fabricante. Durante un escaneo, el láser dispara cada cierto número de grados un haz de luz para estimar la distancia a partir de ese punto.

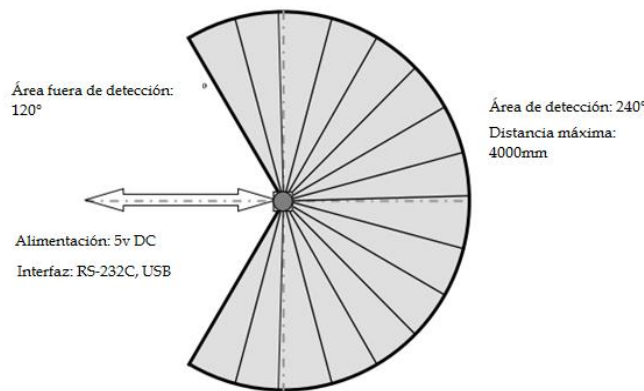


Figura 5.5. Especificaciones técnicas de un láser Hokuyo

Dentro del API que trabaja con los sensores láser, éste devuelve un arreglo de varios elementos de acuerdo al número de muestras que se tomen por escaneo. Cada muestra tiene un ángulo de separación uniforme que llamaremos α . Cada posición del arreglo contiene la distancia medida hasta un objeto próximo en el láser, siendo una distancia infinito cuando no se puede detectar algo más cerca de 4 metros para el caso particular de esta línea de sensores.

Una vez que se obtienen las posiciones, en caso de ser diferente a infinito, es posible calcular la posición relativa espacial con coordenadas en un plano 2D. Para ello, podemos asumir que el ángulo varia de -120° a 120 grados, y el ángulo 0 corresponde al eje de las abscisas (Y). Entonces obtenemos que:

$$x = distancia * \sin(\alpha)$$

$$y = distancia * \cos(\alpha)$$

Ahora, esta posición necesita ser interpretada para localizarla de manera absoluta en el espacio. Esto se realiza mediante matrices de transformación, tomando en cuenta que cada robot tiene un sistema de coordenadas propio que difiere del sistema del plano. Se utilizan matrices de transformación para regresar cualquier coordenada a coordenadas absolutas dentro del espacio.

5.4 PROPUESTA DE MAPEO COOPERATIVO DE ESPACIOS

Para la tarea a resolver en la arquitectura, se decidió realizar una exploración con un mapeo sencillo incluido, a través de las técnicas descritas arriba. Para ello, en vez de realizar un mapeo simple en 2D, se piensa dar profundidad a los objetos.

Si se consideran las capacidades físicas de los robots móviles con los que se cuentan, sus sensores láser únicamente son capaces de detectar objetos en un ambiente 2D, por lo que una mejor vista del mapa la podría dar otro agente que esté a diferente altura que pueda descifrar las alturas de los objetos. Igualmente, ya que es peligroso para un agente móvil aéreo desplazarse a alturas bajas sin poder contar con sensores de proximidad en los ejes de ordenadas y abscisas, es mejor aprovechar la capacidad de sensar sobre el eje de profundidad y al final poder realizar un mapeo cooperativo entre varios agentes del mismo entorno.

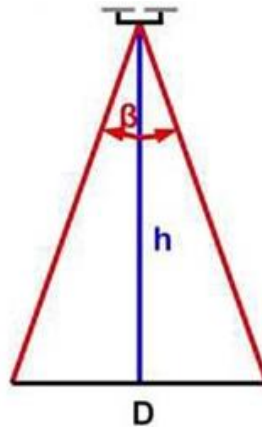


Figura 5.6. Modelo de sensado de altura por un dron

Al final se debe obtener la aproximación de un mapa utilizando las bases de la arquitectura, con información que se comparte de todos los sensores de cada uno de los agentes. Una de las propuestas de esta tarea y de la arquitectura en general es que es posible tener una

capacidad de sensado mayor debido a que la información de todos los sensores es compartida, lo que se denominó como *sensor pool*. Los comportamientos de los robots, el tiempo de mapeo de espacios y la eficiencia de la comunicación nos dan parámetros acerca de la mejor configuración para esta tarea en específico, mientras que también muestran la eficiencia de la arquitectura.

Se realizaron pruebas en un entorno simulado para probar el comportamiento de diferentes robots de acuerdo a diferentes configuraciones de agentes trabajando en el mismo entorno. Por cuestiones de redundancia sin solución, no se colocó más de un agente aéreo en las pruebas realizadas.

5.4.1 Pruebas en entorno simulado

Para el entorno simulado, se realizaron pruebas en dos tipos de escenarios, con diferentes configuraciones de robots, siendo 3 robots móviles terrestres el número máximo reportado en el presente trabajo, y un robot aéreo navegando por encima del entorno.

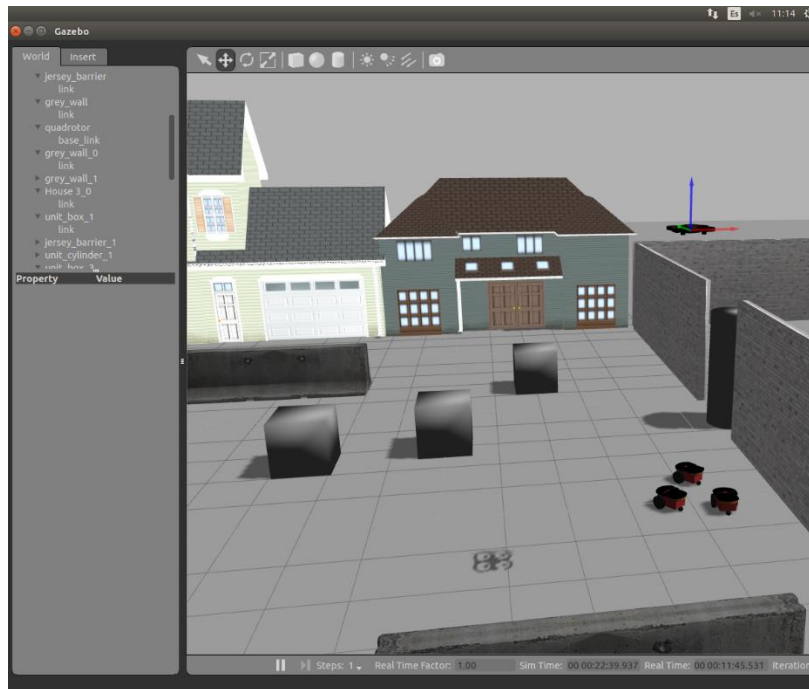


Figura 5.7. Entorno de pruebas del ambiente 1

En todos los casos, el entorno mantiene el mismo tamaño, y lo único que cambia es la configuración de los obstáculos dentro de él. La métrica de comparación para determinar la eficiencia de una prueba es la precisión del mapeo en un tiempo determinado (5 minutos para todas las pruebas), que es comparado contra el mapa real y se realiza una interpolación de los puntos.

Cabe aclarar que todas las pruebas cuyo valor de precisión fue menor al 40% fueron desechadas debido a que se consideraron intentos fallidos en la arquitectura. El número de pruebas realizadas para todos los casos es de 6, bajo las mismas configuraciones para obtener el promedio de la precisión en cada caso.

Agentes terrestres	Agentes aéreos	Precisión	Pruebas fallidas
1	1	67%	1
2	1	64%	2
3	1	47%	4

Tabla 5.1. Tabla comparativa entre promedios de precisión para el ambiente 1

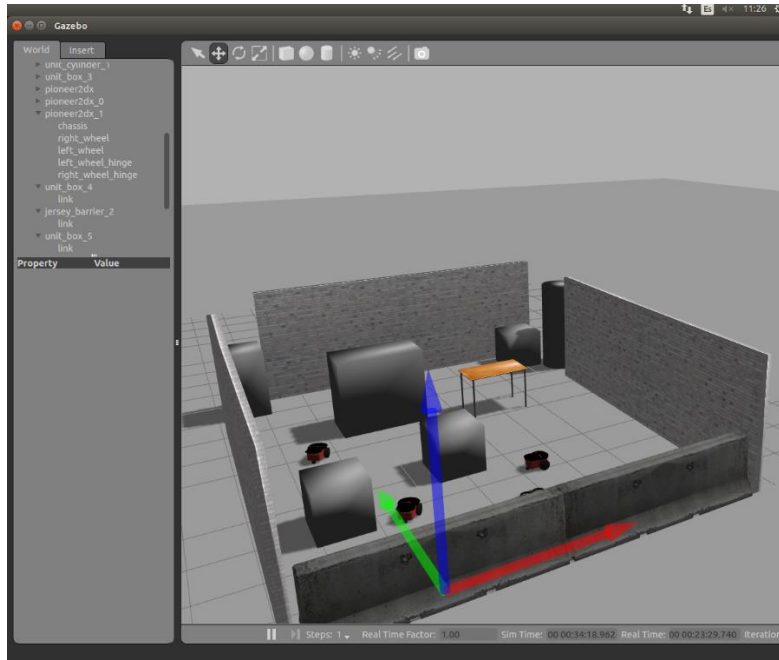


Figura 5.8. Entorno de pruebas del ambiente 2

Agentes terrestres	Agentes aéreos	Precisión	Pruebas fallidas
1	1	73%	1
2	1	60%	1
3	1	43%	3

Tabla 5.2. Tabla comparativa entre promedios de precisión para el ambiente 2

A continuación, se presenta un gráfico comparativo que permite ver la tendencia según el número de robots móviles terrestres que conviven en el mismo entorno.

Algo de lo que se puede observar es que el rendimiento de la tarea va disminuyendo de acuerdo al número de agentes que se tenga, aunque esto tenga que ver con diferentes aspectos no relevantes para la arquitectura, tales como la programación correcta del algoritmo, las posiciones de inicio de los agentes, entre otros datos.

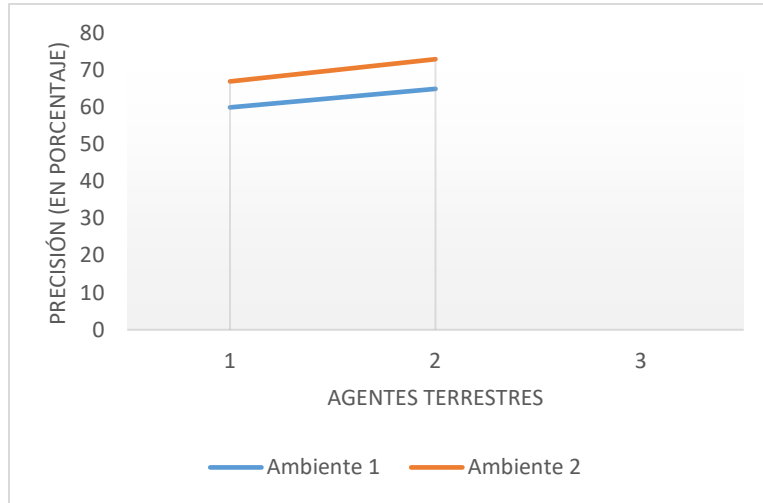


Figura 5.9. Gráfico comparativo de la precisión en diferentes pruebas

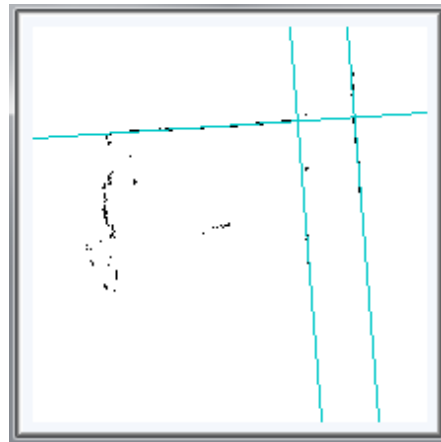


Figura 5.10. Muestra de mapeo en entorno simulado (ambiente 1)

5.4.2 Pruebas en entorno físico real

Para el entorno físico real, se realizaron pruebas en los mismos dos tipos de escenarios, lo más cercano posible a como se configuraron en el simulador, con las mismas configuraciones que las pruebas simuladas, exceptuando que aquí únicamente se cuenta con un robot móvil terrestre y un robot móvil aéreo, por lo que la comparación es entre resultados de pruebas físicas reales vs pruebas simuladas.

En todos los casos, el entorno mantiene el mismo tamaño, y lo único que cambia es la configuración de los obstáculos dentro de él. La métrica de comparación para determinar la eficiencia de una prueba es la precisión del mapeo en un tiempo determinado (5 minutos y 10 minutos), que es comparado contra el mapa real y se realiza una interpolación de los puntos. Se comparan los promedios de las precisiones entre los entornos simulados y reales correspondientes.

Minutos corriendo	Precisión de prueba física real	Precisión de prueba simulada
-------------------	---------------------------------	------------------------------

5	60%	67%
10	77%	83%

Tabla 5.3. Tabla comparativa entre promedios de precisión para el ambiente 1

Minutos corriendo	Precisión de prueba física real	Precisión de prueba simulada
5	65%	73%
10	79%	89%

Tabla 5.4. Tabla comparativa entre promedios de precisión para el ambiente 2

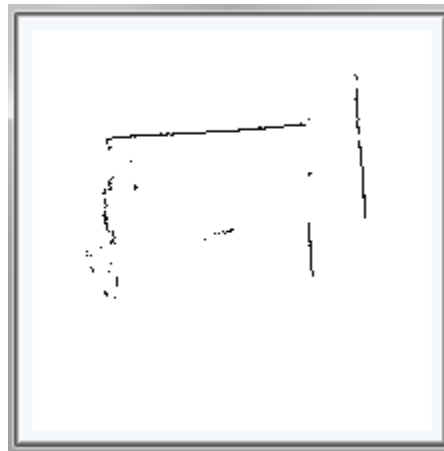


Figura 5.11. Muestra de mapeo en entorno real

Una de las observaciones es que no se obtuvo la misma precisión en un entorno simulado contra el entorno físico real. La principal razón por la que se cree que se obtuvo este resultado es por factores externos al manejo de la tarea y de la arquitectura, tales como la imprecisión y la acumulación de errores que se explica al inicio de este capítulo, ya que se utiliza la navegación por estimación para conocer la posición en la que el robot se encuentra en el espacio.

5.5 CAMBIOS EN LA ARQUITECTURA DE COORDINACIÓN

Algunos cambios tuvieron lugar dentro de la arquitectura para poder ver si se realizaban las tareas de forma eficiente. La mayoría de ellos tuvo resultados similares o peores, pero ellos iban ligados a la tarea que se intenta realizar, por lo que la arquitectura debe ajustarse ligeramente a las necesidades específicas de cada tarea.

Los ajustes que empeoraron el desempeño de la arquitectura para la tarea fueron:

- Utilizar mensajes directos para la comunicación (ya que no eran necesarios).
- Utilizar un número de agentes dinámico (agregar o retirar agentes durante la ejecución de las tareas).

Los ajustes que tuvieron resultados similares a los presentados en la sección anterior fueron:

- Cambiar la coordinación de acciones de planificación a una coordinación reactiva
- Agregar una política de negociación (esto dificultó más las pruebas por lo que se abandonó la idea al no haber mejoría).

6 CONCLUSIONES Y TRABAJO FUTURO

Durante el desarrollo del trabajo, varias observaciones importantes fueron suscitándose, desde la investigación en fuentes como en la planeación y desarrollo de la arquitectura. Se presentan las más relevantes a continuación.

- La mayoría de los trabajos relacionados a sistemas multi robot y multi agente implementan agentes robóticos con las mismas características (sistemas homogéneos).
- En algunos casos, un solo robot puede ejecutar mejor las tareas de lo que lo haría un grupo de robots, de acuerdo a las capacidades que alcance el individuo con respecto a las capacidades de un grupo de robots.
- El comportamiento de un agente está fuertemente ligado a las capacidades que se le quieran dotar (o las que la construcción física del mismo permita) y también a la tarea que se desee resolver.
- El control cooperativo se complica de acuerdo a la movilidad que tengan los agentes, puesto que a controles individuales más complejos mayor anticipación por parte de otros agentes debe existir.
- Se prefieren agentes en los trabajos relacionados con capacidades simples (relación costo-beneficio).
- Muchos algoritmos bio inspirados implementados en los sistemas multi agente representan comportamientos observados y estudiados en conjuntos y hordas de animales.
- La comunicación está fuertemente ligada al tipo de capacidades físicas con las que cuenten los agentes robóticos.
- En los entornos simulados, la comunicación no es muy importante como lo es en los entornos reales físicos.
- Un entorno físico real exige una mayor tolerancia a errores, sobre todo en la comunicación y a errores de navegación, sobre todo si se utiliza una navegación por estimación.
- La coordinación está ligada a las características que los agentes posean.
- Los métodos y agentes reactivos son preferidos sobre los deliberativos debido a la simplicidad de su implementación. Los agentes deliberativos se destinan a tareas más complejas.
- Un número elevado de agentes interactuando en el mismo entorno puede suscitar interferencias causadas por ellos mismos que entorpecen las tareas, y no es posible resolver estas interferencias a nivel de la arquitectura.

Lo que las pruebas nos dan a entender, tanto las que se aplicaron a las partes de la arquitectura como al desempeño de la tarea a resolver es que el número de agentes influye de una manera fuerte. Tanto que el desempeño se degrada conforme se agregan más agentes a la escena, sin embargo esto es debido a la naturaleza de las tareas y del cómo son ejecutadas.

La estrategia de comunicación con pizarra tiende a tener un comportamiento lento si se incrementa el número de robots en el ambiente, esto se debe a que la estructura usada como pizarra es compartida y solo un robot a la vez puede tener acceso a ella, si más de un robot desea acceder a la pizarra, es necesario esperar en una cola por un turno. Por esto se cree que al incrementar el número de agentes el desempeño también se ve afectado.

Igualmente, si no existe una manera de comunicar a los agentes, se puede llegar a resolver la tarea pero se tendría un mayor número de conflictos, lo que a veces podría detener, en el peor de los casos, el progreso de las tareas. Esto deriva a un pobre rendimiento en el sistema, además de ser posible que no se cumpla el objetivo principal.

Desde un punto de vista particular, el desarrollo de este trabajo ayuda a profundizar conocimientos tanto en el campo de la Inteligencia Artificial como en el de la Robótica móvil, ya que para su desarrollo fue necesario estudiar y tomar de ambas lo que mejor contribuya a este tema.

En general desarrollar trabajos como este contribuye al desarrollo científico y tecnológico de México, haciendo crecer la robótica móvil en este país y contribuyendo al crecimiento y uso de la tecnología en áreas como medicina, educación, industria, hogar, seguridad por mencionar algunas.

Finalmente, se concluye que al integrar una estrategia de comunicación a los robots como forma de coordinación, para evitar conflictos, es de gran utilidad en la tarea de exploración de ambientes desconocidos, ya que por la información compartida entre ellos, se evitan conflictos relacionados con colisiones y en algunos casos se evita volver a explorar zonas.

De forma particular podemos afirmar que una buena estrategia de comunicación para la coordinación en la tarea de exploración de ambientes desconocidos contribuye a la reducción del tiempo de exploración. Igualmente se debe considerar que si se resuelven otras tareas, es muy probable que no todas las técnicas funcionen para todos los casos.

6.1 TRABAJO FUTURO

Con la realización de este trabajo, se incluyó una parte muy importante en el laboratorio MOVIS de la Facultad de Ciencias de la Computación de la Benemérita Universidad Autónoma de Puebla. Dicha parte es la inclusión de un framework de desarrollo robótico robusto, en este caso ROS. Con ello se evitó realizar mucho del trabajo no relacionado directamente al desarrollo de la arquitectura. Lo que se pretende ahora es instruir a todos los alumnos en el uso de esta tecnología para que todos los desarrollos futuros en la rama de la robótica sean estandarizados y uniformes, por lo que el seguimiento de los proyectos futuros sea más sencillo.

Con respecto al trabajo, una parte importante que se debe explorar es la realización de navegación y mapeo conjunto utilizando técnicas más precisas que no se basen en la estimación, lo que haría necesario el uso de sensores tipo GPS. Esto facilitaría mucho la precisión en los mapeos y daría mejores resultados. Sin embargo, se debe considerar que los sensores láser y ultrasónicos no funcionan del todo bien en entornos abiertos donde los sensores GPS se desempeñan mejor.

De igual manera se puede evolucionar el mapeo primitivo utilizado aquí por técnicas más sofisticadas conocidas como SLAM (Localización y Mapeo Simultáneos), para hacer más eficiente la tarea a resolver. Incluso puede considerarse otro tipo de tareas a resolver que no tengan que ver con localización explícitamente, tales como seguimiento de personas a través de cámaras, u otros objetivos.

APÉNDICE A TIPOS DE MENSAJE EN EL PROTOCOLO DE COMUNICACIÓN

En esta sección se describen con detalle todos los mensajes utilizados a través de la comunicación dentro de la arquitectura. Como se había mencionado anteriormente, existen tres grupos principales de mensajes: Organización del sistema, Información personal y Llamada a procedimiento remoto. A continuación se describe la estructura genérica de un mensaje, con los campos comunes que se encontraran en todos los mensajes.

Nombre de campo	Tipo de dato	Descripción
Emisor	String	Campo descriptivo que permite identificar al robot de los demás, puede ser un ID establecido por el diseñador.
Tipo de mensaje	Int	Un identificador para saber qué tipo de mensaje se trata.
Campo específico	Estructura de datos	Campo que contiene el mensaje específico
Tiempo	Time	La hora/fecha en la que se envía el mensaje.

Tabla A.1. Estructura genérica de un mensaje

Los campos son informativos sobre quién manda el mensaje y el tipo de mensaje del que se trata, para que el agente central de la comunicación se encargue de verificar que los mensajes están siendo tratados de manera adecuada. Existen campos específicos dentro de los mensajes que permiten describir lo que se está dando a conocer, o en caso de las peticiones, lo que se está solicitando.

A continuación se incluyen los tipos de mensaje con los campos que los conforman, además de una breve descripción de la utilidad de cada campo. En el caso de no ser necesario un campo adicional, se especifica el campo Nulo.

Saludo

Nombre de campo	Tipo de dato	Descripción
Nulo	Empty field	Campo vacío

Tabla A.2. Campos particulares para mensaje Saludo

Keep-alive

Nombre de campo	Tipo de dato	Descripción
Nulo	Empty field	Campo vacío

Tabla A.3. Campos particulares para mensaje Keep-alive

Desconexión

Nombre de campo	Tipo de dato	Descripción
Nulo	Empty field	Campo vacío

Tabla A.4. Campos particulares para mensaje Desconexión

Asignación de tareas

Nombre de campo	Tipo de dato	Descripción
ID de tarea	int	Identificador para las tareas que están programadas dentro de la arquitectura
Tarea cooperativa	boolean	Identifica si la tarea es cooperativa
Num_colaboradores	int	Contiene el número de colaboradores (exceptuando a sí mismo)
Colaboradores	int (array)	Contiene los identificadores de los agentes colaboradores en dicha tarea

Tabla A.5. Campos particulares para mensaje Asignación de tareas

Finalización de tareas

Nombre de campo	Tipo de dato	Descripción
ID de tarea	int	Identificador para las tareas que están programadas dentro de la arquitectura

Tabla A.6. Campos particulares para mensaje Finalización de tareas

Comando de movimiento

Nombre de campo	Tipo de dato	Descripción
Velocidad lineal	float (array)	Contiene un arreglo de tres posiciones indicando velocidad lineal en x , y y z , en ese orden.
Velocidad angular	float (array)	Contiene un arreglo de tres posiciones indicando velocidad angular en x ($roll$), y ($pitch$) y z (yaw), en ese orden.

Tabla A.7. Campos particulares para mensaje Comando de movimiento

Mensaje de confirmación

Nombre de campo	Tipo de dato	Descripción
Nulo	Empty field	Campo vacío

Tabla A.8. Campos particulares para mensaje Mensaje de confirmación

Posición espacial

Nombre de campo	Tipo de dato	Descripción
Punto 3D	float (array)	Contiene un arreglo de tres posiciones indicando coordenadas geométricas espaciales en x , y y z , en ese orden.

Tabla A.9. Campos particulares para mensaje Posición espacial

Información sensorial

Nombre de campo	Tipo de dato	Descripción
Tipo de sensor	int	Indica el tipo de sensor que se está leyendo
Ángulo mínimo	float	Ángulo de inicio del escaneo
Ángulo máximo	float	Ángulo de fin del escaneo
Incremento de ángulo	float	Distancia angular entre mediciones
Incremento de tiempo	float	Tiempo entre mediciones
Tiempo de escaneo	float	Tiempo entre escaneos
Distancia mínima	float	Valor de la distancia mínima en las mediciones
Distancia máxima	float	Valor de la distancia máxima en las mediciones
Distancias	float (array)	Valores de las distancias en las mediciones (tamaño variable calculado entre la apertura del ángulo y el incremento).

Tabla A.10. Campos particulares para mensaje Información sensorial

Estado operativo

Nombre de campo	Tipo de dato	Descripción
Estado	int	Indicador sobre el estado actual del robot

Tabla A.11. Campos particulares para mensaje Estado operativo

Petición de información remota

Nombre de campo	Tipo de dato	Descripción
Propiedad	int	Identificador de la propiedad del robot que se está solicitando

Tabla A.12. Campos particulares para mensaje Petición de información remota

Petición de procedimiento remoto

Nombre de campo	Tipo de dato	Descripción
Procedimiento	String	Nombre del procedimiento que se desea ejecutar en el robot
Parámetros	Object (array)	Parámetros que se solicitan en el procedimiento (en caso de haberlos)

Tabla A.13. Campos particulares para mensaje Petición de procedimiento remoto

BIBLIOGRAFÍA

- [1] I. Struck, «All On Robots,» 2013. [En línea]. Available: <http://www.allonrobots.com/types-of-robots.html>. [Último acceso: 02 05 2017].
- [2] R. A. Brooks, «A Robust Layered Control System for a Mobile Robot,» *Massachusetts Institute of Technology*, 1985.
- [3] I. Dutta, A. D. Bogobowicz y J. J. Gu, «Collective Robotics - A Survey of Control and Communication Techniques,» *Proceedings of the 2004 International Conference on Intelligent Mechatronics and Automation*, pp. 505-510, 2004.
- [4] K. Hernández, *Estrategias de coordinación para la exploración con multi-agentes robóticos*, Puebla: BUAP, 2010.
- [5] I. Navarro y F. Matía, «An Introduction to Swarm Robotics Robotics,» *ISRN Robotics*, vol. 2013, pp. 1-10, 2013.
- [6] R. C. Kube y H. Zhang, «Collective robotics: From social insects to robots,» *Adaptive Behavior*, vol. 2, pp. 189-218, 1993.
- [7] R. C. Arkin, T. Balch y E. Nitz, «Communication of Behavioral State in Multi-agent Retrieval Tasks,» *Proceedings of IEEE International Conference on Robotics and Automation.*, pp. 588-594, 1993.
- [8] V. Dupourque, H. Guiot y O. Ishacian, «Towards multi-processor and multi-robot controllers,» *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 864-870, 1986.
- [9] T. Fukuda y S. Nakagawa, «A dynamically reconfigurable robotic system (concept of a system and optimal configurations,» *Proceedings of IECON*, pp. 588-595, 1987.
- [10] D. Vallejo, P. Remagnino, D. N. Monekosso, L. Jiménez y C. González, «A Multi-agent Architecture for Multi-Robot Surveillance,» *Lecture Notes in Computer Science*, vol. 5796, pp. 266-278, 2009.
- [11] H. Asama, A. Matsumoto y Y. Ishida, «Design of an autonomous and distributed robot system: ACTRESS,» *Proceedings of IEEE RSJ International Workshop on Intelligent Robots and Systems*, pp. 283-290, 1989.
- [12] S. Bonner y K. G. Shin, «A Comparative Study of Robot Languages,» *IEEE Computer*, vol. 12, n° 15, pp. 82-96, 1982.

- [13] Y. U. Cao, A. S. Fukunaga, A. B. Kanhg y F. Meng, «Cooperative mobile robotics: Antecedents and directions,» *Proceedings of the IEEE/RJS International Conference on Intelligent Robots and Systems*, pp. 226-234, 1995.
- [14] L. E. Parker, «Current state of the art in distributed robot systems,» *Distributed Autonomous Robotic Systems*, n° 4, pp. 3-12, 2000.
- [15] G. Dudek, M. Jenkin, E. E. Milios y D. Wilkes, «A taxonomy for multiagent robotics,» *Autonomous Robots*, vol. 4, n° 3, pp. 375-397, 1996.
- [16] M. Wooldridge, «Intelligent Agents,» de *Multiagent Systems: A Modern Approach to Distributed Modern Approach to Artificial Intelligence*, Cambridge, Massachusetts, The MIT Press, 1999, pp. 27-78.
- [17] F. Lewis, H. Zhang, K. Hengster-Movric y A. Das, *Cooperative Control of Multi-Agent Systems*, Springer-Verlag London, 2014.
- [18] J. Liu y J. Wu, *Multi-agent Robotic Systems*, Boca Raton, Florida: CRC Press, 2001.
- [19] M. J. Mataric, «Behavior-based robotics as a tool for synthesis of artificial behavior and analysis of natural behavior,» *Trends in Cognitive Science*, vol. 2, n° 3, pp. 82-87, 1998.
- [20] G. S. Sukhatme y M. M. J., «Embedding robots into the Internet,» *Communications of the ACM - Special Issue on Embedding the Internet*, vol. 43, pp. 67-73, 2000.
- [21] L. E. Parker, *Heterogeneous Multi-Robot Cooperation*. Ph.D. Thesis, USA: MIT, 1994.
- [22] C. Versino y L. M. Gambardella, «Ibots learn genuine team solutions,» *Proceedings of the European Conference on Machine Learning, Lecture Notes in Artificial Intelligence*, vol. 1224, pp. 298-311, 1997.
- [23] D. Floreano y J. Urzelai, «Evolution and learning in autonomous robotic agents,» *Bioinspired Computing Systems: Towards Novel Computational Architectures*, pp. 1-36, 1998.
- [24] A. Farinelli, L. Iocchi y D. Nardi, «Multirobot systems: a classification focused on coordination,» *IEEE Transactions on Systems, Man and Cybernetics*, vol. 34, n° 5, pp. 2015-2018, 2004.
- [25] M. Woolridge y M. Wooldridge, *Introduction to multiagent systems*, New York: John Wiley & Sons, Inc, 2001.
- [26] N. Nguyen, R. Kowalczyk y S.-M. Chen, «Multi-agent Architecture for Multi-robot Surveillance,» *ICCCI 2009, LNAI 5796*, pp. 266-278, 2009.

- [27] A. K. Chopra y M. P. Singh, «Agent Communication,» de *Multiagent Systems*, MIT Press, 2011.
- [28] B. P. Gerkey y M. J. Mataric, «Are (explicit) multi-robot coordination and multi-agent coordination really so different?,» *Procedures of the AAAI Spring Symposium on Bridging the Multi-Agent and Multi-Robotic Research Gap*, pp. 1-3, 2004.
- [29] D. Greenwood, F. Bellifemine y B. Caire, *Developing multi agent systems with jade*, San Diego, CA: WILEY, 2004.
- [30] J. Liu y J. Wu, *Multi-Agent Robotic Systems*, Florida: CRC Press, 2001.
- [31] W. Burgard, M. Moors, D. Fox y R. T. S. Simmons, «Collaborative multi-robot exploration,» *Proceedings of the IEEE International Conference on Robotics and Automation*, 2000.
- [32] A. Bicchi, L. Pallottino, V. G. Scordio y E. Frazzoli, «Decentralized Cooperative Policy for Conflict Resolution in Multivehicle Systems,» *IEEE Transactions on Robotics*, vol. 23, nº 6, pp. 1170-1183, 2007.
- [33] S. Russell y P. Norvig, *Artificial Intelligence: A Modern Approach*, Prentice-Hall, 1995.
- [34] G. Weiss, *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, Cambridge: MIT Press, 1999.
- [35] L. Zadeh, «Fuzzy logic = computing with words,» *IEEE Transactions on Fuzzy Systems*, vol. 4, nº 2, pp. 103-111, 1996.