

Metodología para el Análisis y Diseño de Sistemas Multi-Agente Robóticos: MAD-Smart

Jovani Alberto Jiménez B.
Profesor de la Escuela de
Sistemas
Universidad Nacional de
Colombia Sede Medellín
jajimen1@unal.edu.co

Marcela Vallejo Valencia
Estudiante de Ingeniería
Electrónica
Universidad de Antioquia.
emavv736@udea.edu.co

John Fredy Ochoa Gómez
Estudiante de Ingeniería de
Sistemas e Informática
Universidad Nacional de
Colombia Sede Medellín.
jfochoa@unal.edu.co

RESUMEN

En este artículo se describe la metodología para el desarrollo de Sistemas Multi-Agente Robóticos MAD-Smart y se desarrollan algunos ejemplos para visualizar algunas de sus características. La metodología busca dar cobertura en la fase de conceptualización, análisis y diseño mediante la definición de un conjunto de actividades que ayudan al desarrollador a entender mejor el problema a resolver, las características finales que deberá tener el sistema y el papel que cada agente juega en la solución del problema.

Categorías

I.2.11 [Inteligencia Artificial Distribuida]: Agentes Inteligentes, Sistemas Multi-Agente.

Términos Generales

Diseño, Estandarización.

Palabras Clave

Sistemas Multi-Agente, Metodologías de Desarrollo de MAS, Entornos Cooperativos Robóticos, Enjambres Robóticos, Sistemas Multi-Agentes Robóticos.

1. INTRODUCCIÓN

Una de las problemáticas de la robótica móvil que más interés ha suscitado en los últimos años es la navegación de agentes robóticos que trabajan cooperativamente en la solución de problemas como el desplazamiento de cargas [1], entrega de materiales en oficinas [12], construcción de mapas de entornos desconocidos [31], entrega de medicinas en hospitales [33], detección de minas [13], entre otros [6] [2].

El trabajo en esta área es lo suficientemente amplio y se han aplicado conceptos de etología animal, de la teoría de la organización, el aprendizaje y de la inteligencia artificial distribuida, siendo de esta última de donde ya han salido algunas propuestas de metodologías para el desarrollo de proyectos [3] [7] [34]. En este punto surgen ciertos problemas ya que a pesar de que los entornos cooperativos robóticos se pueden abordar desde la teoría de Sistemas Multi-Agente, los agentes robóticos tienen características diferentes a los agentes computacionales [14]. Adicionalmente las técnicas de coordinación y cooperación en los sistemas multi-agente (MAS) no son las más adecuadas para el tratamiento de incertidumbre y falta de información que hay comúnmente en la robótica [34]. Algunos de los requerimientos que impone el funcionamiento de agentes en entornos reales son los siguientes [24]:

- Saberse comportar de acuerdo a las circunstancias (situated behavior), y reaccionar ante sucesos imprevistos.

- Las tareas deben realizarse con eficiencia teniendo en cuenta las imposiciones restrictivas del trabajo en tiempo real.

- Se debe tener en cuenta la presencia de otros agentes en el entorno.

El papel principal de una metodología es permitir transformar la visión subjetiva de un sistema en una definición objetiva que permita su implementación. De esta manera, una metodología debe de ofrecer [8]:

- Un conjunto de pasos con actividades a desarrollar e instrucciones para desplazarse entre los diferentes pasos.

- Mecanismos de documentación que permitan compartir la experiencia ganada en el proyecto con otros desarrolladores e investigadores.

- Terminología homogénea que brinde significado a los pasos del ciclo y facilite la transición entre pasos (por lo general es una terminología gráfica basada en diagramas de flujo).

- Estructuras conceptuales abstractas que permitan una suficiente elección de técnicas cuando se deba de implementar el sistema (independencia de técnicas de implementación).

En el desarrollo de metodologías para el diseño de MAS, los investigadores han enfocado sus esfuerzos en extender las metodologías existentes. Estas extensiones se han realizado principalmente sobre dos áreas: Metodologías orientadas a objetos (Object Oriented) e ingeniería de conocimiento (Knowledge Engineering) [17]. La metodología propuesta extiende a su vez de las metodologías para MAS existentes, buscando dar mayor cobertura a la implementación física y al desarrollo de estrategias para la solución cooperativa de problemas.

El artículo está distribuido de la siguiente manera: en la sección dos se introduce la metodología explicando sus orígenes y concretando algunos términos claves, en la sección tres se muestran como se documentan algunos aspectos de proyectos de Sistemas Multi-Agentes Robóticos, la sección cuatro discute algunos aspectos de la metodología y finalmente en la sección cinco se sacan algunas conclusiones respecto a las expectativas de la metodología y se discuten algunos aspectos que aún quedan por abordar.

2. METODOLOGÍA MAD-SMART

El enfoque de la Metodología para el Análisis y Diseño de Sistemas Multi-Agentes Robóticos MAD-Smart esta

fundamentado en: (1) Independencia de las técnicas de implementación, (2) Proceso metodológico ascendente iniciando en la determinación de los requerimientos del proyecto para llegar a las estrategias de solución de éstos de manera cooperativa.

La metodología recoge elementos de trabajos existentes para la construcción de Sistemas Multi-Agente como son GAIA [37], Mas-CommonKADS [18] y MaSe [9] a la vez que introduce otros nuevos que surgen desde la propia área de los Sistemas Multi-Agentes Robóticos. Hay que tener en cuenta que las metodologías surgen en gran medida para asesorar el desarrollo a nivel industrial de una tecnología [28], por lo que el desarrollo de Sistemas Multi-Agente Robóticos al realizarse a tan baja escala se encuentra desprovisto de metodologías de aceptación general.

Se debe de tener en cuenta que la metodología no esta orientada hacia la síntesis automática de grupos equipos. Este campo hasta ahora se ha abordado en su mayor parte desde la computación evolutiva permitiendo la definición de estrategias a partir de un grupo de robots heterogéneos y una misión para llevar a cabo [29]. En lugar de esto la metodología propuesta se busca que sirva como herramienta de documentación y guía de proyectos a la vez que permite conservar conocimiento que puede ser reutilizado en desarrollos posteriores.

Los ocho pasos que constituyen la metodología (Figura 1) están orientados a capturar la definición de mecanismos para la comunicación y manejo de conceptos entre los agentes, la descripción de los agentes individuales –incluyendo características hardware y software–, la descripción de los mecanismos de interacción entre los miembros del grupo, y la distribución de tareas entre los agentes.

2.1 Terminología.

Algunos de los términos que serán usados a lo largo de las actividades de la metodología son los siguientes:

- Agente: Entidad física o abstracta que puede percibir su ambiente mediante sensores, es capaz de evaluar tales percepciones y tomar decisiones por medio de mecanismos de razonamientos simples o complejos, comunicarse con otros agentes para obtener información y actuar sobre el medio en el que se desenvuelven a través de ejecutores.

- Comportamiento: Constituye una regla de control que permite alcanzar o mantener un objetivo particular.

- Interacción: Es una relacional física o funcional que se da entre dos o mas agentes y que esta mediada por comportamientos bien definidos.

- Comportamiento Colectivo: Denota el comportamiento de una agente en un sistema en el cual hay mas de un agente.

- Comportamiento Cooperativo: Es una subclase de comportamiento colectivo que esta caracterizado por la asociación de partes en busca del beneficio mutuo.

Un sistema multi-agente robótico mostrara comportamiento cooperativo si, debido a algún mecanismo de cooperación, hay un incremento en el desempeño del sistema. El mecanismo de cooperación puede estar basado en la imposición por parte del diseñador de una estructura de control o comunicación, en aspectos de especificación de tareas, en las dinámicas de interacción de los comportamientos de los agentes [4].

- Rol: Es la descripción abstracta de una funcionalidad que se espera satisfaga algún objetivo importante en el sistema. Un agente en su periodo de ejecución puede desempeñar uno o más roles. La notación para especificar el ciclo de ejecución de un rol hace uso de los siguientes elementos (tabla 1):

Tabla 1. Esquemas de ejecución de los roles. x e y son dos actividades (o agrupaciones de actividades) cualquiera

Notación	Descripción
$x.y$	Se realiza la actividad x seguido de y
x^*	x se ejecuta 0 o mas veces
x^w	x ocurre infinitas veces
$x y$	x se ejecuta en paralelo con y
$x y$	Se ejecuta x o y
x^+	x ocurre 1 o mas veces
$[x]$	x es opcional

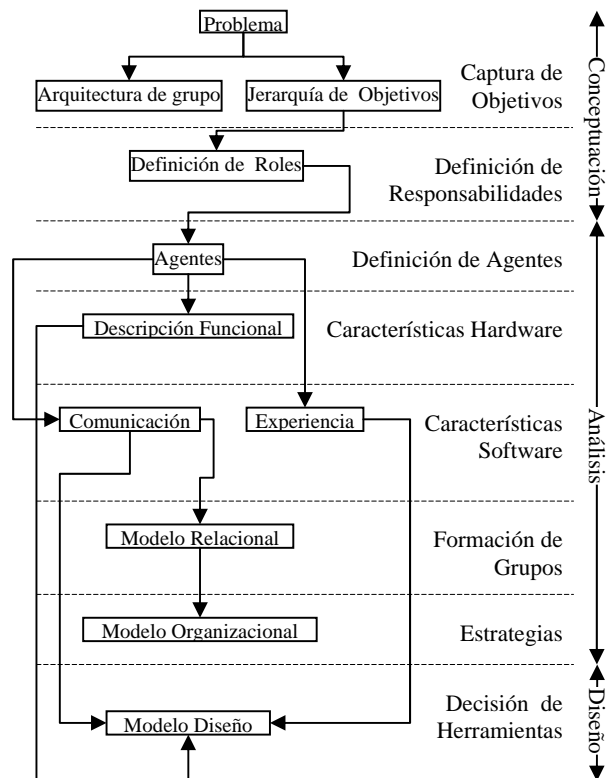


Figura 1. Actividades de la Metodología MAD-Smart

- Interferencia y Conflicto: Interferencia hace alusión a alguna influencia que no permite que un agente alcance un objetivo. Se puede competir por un recurso limitado como el espacio o por un recurso requerido por varios agentes en un momento determinado para alcanzar objetivos diferentes.

- Comunicación: Es el modo mas común de interacción entre agentes. En estos sistemas puede haber comunicación directa o mediante el paso explícito de mensajes y comunicación indirecta o mediante el sentido de otros agentes o el medio.

Otros términos relevantes de la metodología se introducirán a medida que se muestren las actividades en las cuales son relevantes.

2.2 Conceptuación

La finalidad de esta etapa es permitir al desarrollador tener una mejor comprensión de cual es el problema a resolver y el alcance y limitaciones del sistema a desarrollar. A este nivel la definición de los agentes no es importante. Los pasos a desarrollar son:

- Descripción verbal del Proyecto: Se busca obtener una descripción completa en lenguaje natural del sistema que se esta buscando haciendo énfasis en los objetivos que se espera que cumpla. Es de utilidad que el encargado de la descripción haga uso de la terminología propuesta de manera que se facilite el desarrollo de los pasos posteriores. Es valido hacer uso de material de otros proyectos a modo de ejemplo.

- Identificación de los objetivos: Los objetivos descritos en la fase anterior se estructuran mediante las relaciones que tienen y los sub-objetivos que engloban. Los objetivos se jerarquizan mediante un árbol en el cual los nodos hijos representan los sub-objetivos en los cuales se puede descomponer el nodo padre.

- Definición de la arquitectura del grupo: La arquitectura de un sistema computacional es aquella parte del sistema que permanece inalterada hasta que un agente externo la cambie [22]. Este modelo permite definir la infraestructura sobre la que se implementaran los comportamientos colectivos y determinar las capacidades y limitaciones del sistema. Se hace uso de una tabla en la cual se nombra el ítem, la decisión y la descripción tomada. Algunos ítems que se definen en esta actividad son:

- *Nombre del Sistema:* Permite denotar la naturaleza del sistema.
- *Toma de Decisiones:* Existe básicamente dos mecanismos: centralizado, un agente determina que tareas deben llevar a cabo los otros agentes; y distribuido, cada agente determina las acciones que llevará a cabo teniendo en cuenta su estado actual y negociaciones con otros agentes.
- *Distribución del Conocimiento:* Igual que el anterior se tiene: centralizado un agente tiene el modelo del mundo, modelo que es actualizado por información que proviene de otros agentes y distribuido en el cual cada agente maneja su propio modelo del entorno y comparte información solo cuando es requerido.
- *Comunicación:* Cuando es por modificaciones del entorno o mediante la interacción de comportamientos bien definidos, se habla de comunicación indirecta. Cuando la comunicación se hace por un canal explicito, como por una banda del espectro electromagnético, mediante el intercambio de mensajes con algún tipo de significado semántico, la comunicación es de carácter directo.
- *Topología de la Comunicación:* Determina como transcurre la información en el sistema. Por ejemplo punto a punto, broadcast.
- *Redundancia:* Los sistemas redundantes son aquellos en los cuales ante el fallo de un integrante, el sistema sigue en operación ya que otro integrante ocupa su lugar sin comprometer el desempeño del sistema.

Otros elementos que se pueden tener en cuenta son [10]: Rango de Comunicación, Ancho de Banda para la comunicación, Mecanismos de Reconfiguración de los equipos, habilidad de procesamiento del equipo.

-Definición de los Roles: Mediante los roles se define las funcionalidades, a un nivel muy abstracto, que permitirán que los objetivos sean alcanzados. El correcto modelado de los roles permitirá la identificación de los agentes en la fase de análisis [21]. Mataric et al, muestra como la distribución de tareas entre los agentes puede conducir a diferentes tipos de sistemas [17].

2.3 Análisis

Al finalizar esta fase se definirán o se reutilizarán los agentes con sus respectivas características físicas, de comunicación, razonamiento, y las estrategias para la formación de equipos y la solución cooperativa de tareas para el alcance de objetivos.

Se define tres capas a desarrollar en el agente con similitud a la propuesta de CASSIOPEIA [17]: Arquitectura Interna: define la operación del agente a nivel individual en condiciones normales; Capa Relacional: Aquí se encuentran los comportamientos necesarios para la formación de grupos de trabajo; Capa Organizacional: Es la encargada de la exhibición de comportamientos para la solución cooperativa de tareas. Las actividades a desarrollar son las siguientes:

- Desarrollo del modelo de Agente: Este modelo permite especificar las características de un agente como son sus capacidades de razonamiento, la manera en que interactúa con el mundo mediante sensores y efectores, y los grupos de agentes a los que pertenece. Se encuentra basado en el modelo de Agente de Mas-CommonKADS [18]. Hay que tener presente que los agentes son en muchos aspectos una abstracción conveniente, la cual puede o no tener inteligencia. El único requisito es que se pueda integrar dentro del ámbito de cooperación del sistema, esto es, que este orientado a satisfacer algún objetivo.

- Desarrollo del Modelo de la Experiencia: En esta actividad se describe el conocimiento requerido por los agentes para alcanzar sus objetivos. De acuerdo a la generalidad que se acepta en la metodología para los agentes estos pueden bien o manejar conocimiento o bien trabajar sobre datos. Al finalizar esta actividad se tendrán definidos:

- *La Ontología del sistema:* Una ontología permite estructurar el conocimiento que se tiene respecto a un dominio especificado en torno a un conjunto de conceptos que son claves en ese dominio [5]. Hay que tener en cuenta que los conceptos no tienen que estar necesariamente dados como información explicita, por ejemplo cierta alteración del entorno por parte de un agente puede representar un concepto en el sistema.
- *Representación del Mundo:* Hay varios modos de representar el mundo, por ejemplo los Vectores de Características, las Representaciones Icónicas [27], o métodos de representación espacial o geométrica [32]. Se desarrollarán tantas representaciones del mundo como versiones hayan para los diferentes agentes, esto debido a que los agentes podrían o no tener un modelo del mundo.
- *Representación de los otros agentes:* El modelado de las intenciones, creencias, acciones, capacidades, y estados de

otros agentes puede conducir a mecanismos más efectivos de cooperación entre robots [4]. Los modelos mas usados son los probabilísticos [15], inferencias sobre invariantes de información [12], aunque el usuario podría definir uno propio. Se desarrollan tantas representaciones de los agentes como modelos manejen los agentes de sus compañeros.

- **Arquitectura Interna de los Agentes:** Constituye la primera capa de un agente. La arquitectura a usar es de libre elección pudiendo ser reactivas, deliberativas, híbridas, BDI, entre otras. Componentes comunes de los agentes son: sistemas de sentido, de representación, de procesamiento y de acción. Los sistemas basados en comportamiento son los que han logrado mayor participación en esta área [25].

- **Desarrollo del modelo de comunicación:** La cooperación es un modo de interacción que se suele encontrar basada en alguna forma de comunicación [25]. Si la comunicación es directa la cooperación es explícita, si la comunicación es indirecta la cooperación es implícita. Al finalizar esta actividad se deberá tener definidos cuales agentes se comunican de manera directa y cuales de forma indirecta –*modelo de conocidos*-, así como los protocolos o especificaciones de pasos de mensajes entre agentes –*modelo de conversaciones*-. Propuestas para estructuras de comunicación se puede encontrar en [11].

- **Desarrollo del modelo Relacional:** En esta fase el diseñador especifica los comportamientos de la Capa Relacional que permiten a los agentes la formación y disolución de grupos de trabajo y documenta los escenarios en los cuales la formación de dichos grupos será posible.

El modelo jerárquico de objetivos permite determinar donde convergen los agentes para la solución de objetivos que van mas lejos de su alcance individual y solo se cumplirán mediante la cooperación de los agentes implicados.

- **Desarrollo del modelo Organizacional:** Una vez conformados los grupos que participaran en la solución de un objetivo, se debe de generar las estrategias de cooperación de forma que el equipo actúe de una manera efectiva. En esta fase también se especifican los comportamientos cooperativos o comportamientos de la capa Organizacional.

.- **Especificación de los Módulos Físicos de las Plataformas:** Teniendo en cuenta toda la información recolectada de las fases anteriores se bosqueja los requerimientos hardware por parte de los agentes. Se busca hacer énfasis en las características físicas que deberán de tener los agentes robóticos finales, aspecto muy relegado en las metodologías existentes [8].

2.4 Diseño

El propósito en esta fase es acercar lo más posible los agentes definidos en la fase anterior a su posterior implementación. En este punto se tiene en consideración para cada agente su parte Hardware y Software en una misma plantilla. El nivel de abstracción manejado se busca que sea lo mas alto posible ya que las posibilidades de implementación en el campo de la robótica son muy amplias tanto a nivel de los sensores y efectores disponibles como a nivel de procesamiento y control [19].

3. Modelado de Sistemas Multi-Agente Robóticos Mediante Mad-Smart.

En esta sección se plantean algunos modelos que permiten ejemplificar varias de las actividades enunciadas en la sección anterior.

3.1 Conceptuación

3.1.1 Actividad: Desarrollo del Modelo Jerárquico de Objetivos

La Figura 2 muestra la descomposición en sub-objetivos de un caso simple de Digitalización de Entornos [23] [20]. Se debe tener una buena relación entre el nivel de abstracción en los objetivos definidos de manera que sean lo mas genéricos posible a la vez que sean alcanzables por el sistema. Los objetivos están numerados para poder asignarlos luego mediante el numero a roles específicos del sistema.

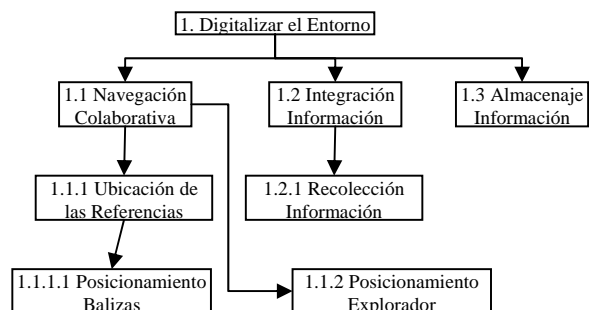


Figura 2. Modelo Jerárquico de Objetivos

3.1.2 Actividad: Desarrollo del Modelo de Arquitectura del Grupo

La tabla 2 muestra la Arquitectura de Grupo básica para un sistema Digitalización Cooperativa de Entornos Estructurados.

Tabla 2. Modelo de Arquitectura de grupo

Ítem	Decisión	Descripción
<i>Nombre del Sistema</i>	Américo	Cartógrafo Italiano
<i>Numero de Plataformas Robóticas</i>	4	Tres balizas y un explorador
<i>Toma Decisiones</i>	Descentralizada	Cada agente toma sus propias decisiones a partir de su estado interno y del estado del medio
<i>Característica del grupo</i>	Heterogéneo	Los agentes baliza se diferencian perfectamente del explorador
<i>Comunicación</i>	Directa	Comunicación explícita mediante RF entre los agentes.
<i>Topología de</i>	Punto a Punto	Todo mensaje

comunicación		tiene un destinatario definido
Redundancia	No	Cada agente juega un papel único en el sistema

3.1.3 Actividad: Desarrollo del Modelo de Roles.

El modelo incluye una descripción del rol propuesto, la condición bajo la cual estará activo en el sistema, la rutina que ejecuta cuando se encuentra activo y los objetivos se espera satisfaga (Figura 3).

Plantilla del Rol: Explorador
Descripción: Este rol se encarga de definir la ubicación del explorador
Condición de Activación: Este rol siempre está activo verificando que el explorador no se detenga en ninguna parte del entorno sin antes haber obtenido la información necesaria.
Actividades: Deambular: X, Sensar el Entorno: Y, Obtener posición actual: Z, Enviar información al Integrador: W.
Objetivos que satisface: 1.1.2 – 1.2.1
Esquema de Ejecución del Rol: (X.Y.Z.W) w

Figura 3. Plantilla Rol Explorador

3.2 Análisis.

3.2.1 Actividad: Desarrollo del Modelo de Agente

La plantilla para la definición de agentes permite especificar, entre otras cosas, los roles que un agente podrá asumir y las capacidades esperadas para que pueda hacerlo (Figura 4).

Agente: A_Explorador
Capacidades de Sensado/Acción: Movimiento a través del entorno, determinación de presencia/ausencia de objetos, comunicación con otros agentes mediante radiofrecuencia, recepción de eco ultrasónico.
Capacidades Razonamiento/Experiencia: Capacidad de desplazamiento libre dentro del ámbito actual de las balizas.
Roles: Explorador
Grupo al que pertenece: Américo.

Figura 4. Plantilla Agente Explorador

3.2.2 Actividad: Desarrollo del Modelo de la Experiencia

En esta actividad se muestran las herramientas de razonamiento de las cuales harán uso los agentes para la solución de sus tareas. La Figura 5 muestra un diagrama de Arquitectura Interna para un agente propuesto por Maja Mataric [26] y la figura 6 muestra un ejemplo de especificación de ontología para un sistema de navegación cooperativa.

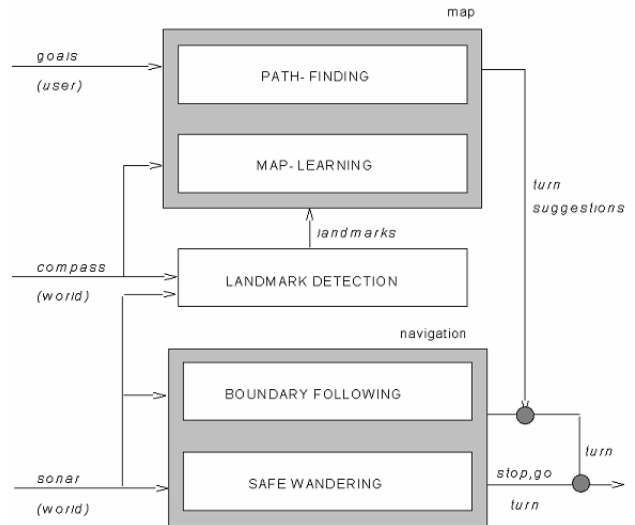


Figura 5. Arquitectura Interna para Navegación propuesta por Mataric.

Concepto: Entorno
Descripción: Medio físico en el que se desenvuelve el sistema
Atributos: Matriz NxM de reales. Cada entrada es el grado de probabilidad de que en esa posición se encuentre un objeto
Concepto: Posición
Descripción: Ubicación de cualquiera de los robots en el entorno
Atributos: Pareja (i,j)
Concepto: Medida_Distancia
Descripción: Última Distancia Medida por los sensores ultrasónicos del explorador
Atributos: Valor real X
Concepto: Medida_Presencia
Descripción: Valor presente en los sensores Infra-Rojos
Atributos: Valor Booleano P

Figura 6. Especificación de una Ontología para un sistema de Navegación Colaborativa en escenarios estructurados

3.2.3 Actividad: Desarrollo del Modelo de Comunicación

-Modelo de Conocidos: Permite especificar gráficamente cuales agentes interactúan. Los sentidos de las flechas indican los iniciadores de las interacciones (Figura 7).

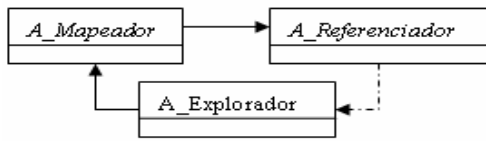


Figura 7. Modelo de Conocidos

-*Modelo de Conversaciones:* Se describe cada una de las secuencias del modelo de conocidos. Un ejemplo de DeLoach para rescate cooperativo de víctimas [7] se puede apreciar en la figura 8.

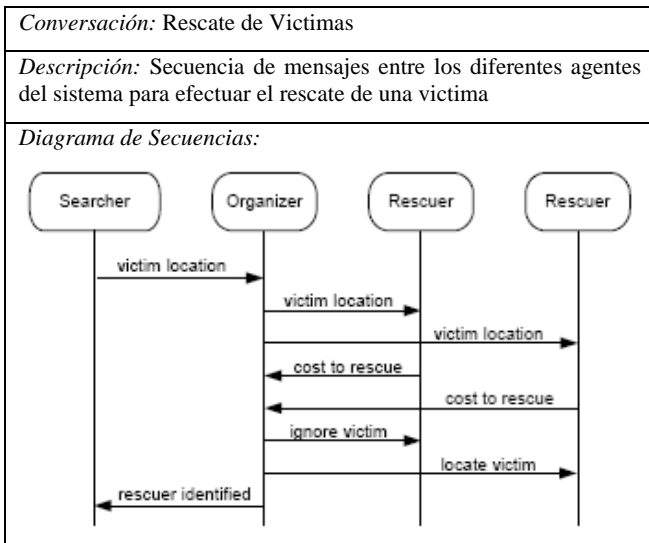


Figura 8. Escenario de Conversación para el rescate de víctimas.

3.2.4 Actividad: Desarrollo del Modelo Relacional

El modelo jerárquico de objetivos permite determinar donde convergen los agentes para la solución de objetivos que van mas lejos de su alcance individual y solo se cumplirán mediante la interacción de los agentes implicados.

-*Desarrollo de los escenarios de formación de grupos:* La figura 9 muestra un escenario de formación de grupo donde el responsable de la formación del grupo es un agente llamado A_Mapeador y donde se asume la disolución del grupo cuando se pierdan los lazos de comunicación entre los agentes.

<i>Nombre del Grupo Formado:</i> Grupo_Mapeador
<i>Agentes que intervienen:</i> A_Mapeador, A_Referenciador, A_Explorador
<i>Responsable:</i> A_Mapeador
<i>Secuencia de Formación del Grupo:</i>
<ol style="list-style-type: none"> 1. En el estado inicial el A_Mapeador se comunica con A_Referenciador y le ordena la posición inicial. 2. A_Referenciador se ubica y da la referencia para A_Explorador. 3. A_Explorador da su primera aproximación del entorno

Grupo Formado
<i>Secuencias Alternativas:</i>
[2.1] A_Referenciador no alcanza la posición deseada pero dicta de igual manera la referencia A_Explorador
[2.2] A_Explorador da la aproximación con coordenadas diferentes a las deseadas
[2.3] A_Mapeador decide si son coordenadas apropiadas, si no lo son dicta nuevas coordenadas, si lo son Grupo Formado
<i>Políticas de Disolución del Grupo:</i>
Perdida de comunicación entre los agentes: A_Mapeador dará un aviso.
Finalización del ciclo de Mapeo: A_Mapeador entregara la versión digital del entorno.

Figura 9. Escenario de Formación de Grupo

-*Descripción de los comportamientos relacionales de los agentes:* Los comportamientos relacionales son aquellos que se dan dentro del marco de formación de grupos de trabajo. El conjunto de estos comportamientos para un agente determinado conforman la Capa Relacional del agente. La figura 10 muestra la descripción de un comportamiento relacional para el un agente que participa en el escenario de la figura 9.

<i>Comportamiento Relacional:</i> enviar_info_entorno
<i>Agente Propietario del Comportamiento:</i> A_Explorador
<i>Condición de Activación:</i> Llegada de la señal de triangulación
<i>Condición de Permanencia:</i> Ninguna
<i>Condición de Finalización:</i> Llegada de la señal de modelo actualizado
<i>Conversaciones:</i> estado_entorno, dictar_posicion

Figura 10. Comportamiento relacional para el A_Explorador

3.2.5 Actividad: Desarrollo del Modelo Organizacional

-*Secuencia de Cooperación:* En este punto se definen las estrategias para el alcance de objetivos. Las estrategias son acciones que los agentes deben de ejecutar para alcanzar objetivos a largo plazo. La figura 11 muestra la descripción de una estrategia en la metodología y la figura 12 muestra un comportamiento cooperativo que exhibe un agente en la estrategia.

<i>Nombre del Grupo Implicado:</i> Grupo_Mapeador
<i>Objetivos a alcanzar:</i> 1 –en el modelo jerárquico de objetivos-
<i>Secuencia de Alcance del Objetivo:</i>
<ol style="list-style-type: none"> 1. En cualquier momento el A_Mapeador da coordenadas de referencia A_Referenciador. 2. A_Referenciador se mueve a las posiciones dictadas 3. A_Referenciador dicta la referencia A_Explorador 4. A_Explorador sensa el entorno 5. A_Explorador envía el estado del entorno A_Mapeador

6. Si A_Mapeador tiene llena la matriz del entorno fin, en otro caso volver a 1.

Figura 11. Estrategia para el mapeo del entorno

<i>Comportamiento:</i> mapeo_entorno
<i>Agente Propietario del Comportamiento:</i> A_Mapeador
<i>Condición de Activación:</i> estado_entorno
<i>Condición de Permanencia:</i> ninguno
<i>Condición de Finalización:</i> ninguno
<i>Efectos:</i> Se actualiza la versión digital que se tenía del entorno.

Figura 12. Plantilla para Comportamiento cooperativo

3.2.6 Actividad: Desarrollo del Modelo de descripción Modular.

Esta actividad permite delimitar ciertas características físicas que se esperan deben de tener las plataformas. Un ejemplo de plantilla es el mostrado en la figura 13.

<i>Sistema Mecánico:</i> La tracción se logra con servomotores Futaba. No hay otros sistemas mecánicos en la plataforma.
<i>Sistema Sensorial:</i> Se cuenta con sensores de contacto para saber cuando se ha tropezado con un objeto. No hay otros sistemas sensoriales.
<i>Sistema de Comunicación:</i> Hay comunicación serial mediante módulos de RF con el A_Mapeador.
<i>Sistema Control:</i> Sistema basado en microcontrolador de 8 bits.

Figura 13. Plantilla para descripción modular

Después de tener claros los módulos que van a constituir al sistema, se clarifican las relaciones entre estos módulos mediante un diagrama de bloques (Figura 14).

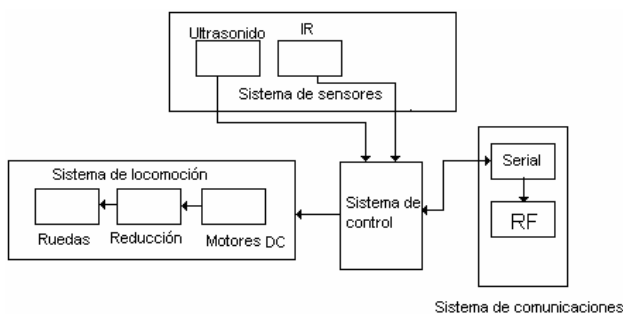


Figura 14. Relación entre los módulos para la plataforma exploradora.

3.3 Diseño.

Se define una plantilla en la cual se especifican ciertas características que tendrán los agentes en su versión funcional. Un ejemplo de plantilla se ve en la figura 15.

<i>Nombre Agente:</i> A_Explorador
<i>Lenguaje Implementación:</i> C

<i>Plataforma Implementación:</i> plataforma exploradora
<i>Subsistemas Software:</i> Ninguno
<i>Comportamientos:</i> Lista de comportamientos obtenidos para los tres niveles a lo largo de la metodología.
<i>Subsistemas Hardware:</i> Mecánicos: Servomotores Futaba s3003 Sensoricos: Sensor infrarrojo sharp GP2D12; Sensor de distancia por ultrasonido SRF10 Comunicación: Modulo transmisor de RF 433Mhz marca LINX TXM-315-LR; Modulo receptor de RF 433Mhz marca LINX RXM-433-LR Control: Microcontrolador de montaje superficial MC68HC908AP64

Figura 15. Plantilla de Diseño para el A_Explorador

4. DISCUSIÓN

4.1 Ventajas de la Metodología

La metodología propuesta trata de dar cobertura a gran número de los problemas que se encuentran en los Sistemas Multi-Agentes Robóticos como son: Naturaleza de la información en el sistema, mediante el modelo de la experiencia; comunicación entre los agentes, modelo de comunicación; y estrategias de cooperación para la solución de problemas mediante los modelos relacional y organizacional.

Otro aspecto relevante de la metodología es que busca desde el principio que la solución del problema sea de naturaleza distribuida mediante un buen modelado de los objetivos, llegándose finalmente a los agentes individuales con sus respectivas características software y hardware.

La metodología esta mejor pensada para el modelado a nivel microscópico en el cual se considera cada robot en el sistema con sus respectivas interacciones y comportamientos en un nivel alto de detalle. Hay que tener en cuenta que la mayor parte del trabajo esta orientado al modelado a nivel macroscópico en el cual se trabaja a nivel de comportamiento del sistema sin consideración de los agentes individuales [36].

Finalmente la similitud con las metodologías orientadas al diseño de Sistemas Multi-Agente lo hace de fácil uso para las personas con experiencia de trabajo en esta área.

4.2 Aspectos no abordados

La metodología no esta pensada para la definición de métricas y de costos de los sistemas individuales (como el uso de ciertos sensores) y de componentes interactivos (como aquellos relacionados con la comunicación).

Tampoco se trabaja la parte de aprendizaje y de solución de conflictos que constituyen líneas básicas de investigación en el área [4].

El manejo de incertidumbre exige también un mayor análisis que el aquí propuesto, aunque el modelo de descripción modular donde se describe la naturaleza de los sensores requeridos y el modelo de diseño donde la escogencia de un sensor determinado se debe de hacer de acuerdo a la naturaleza de la información

requerida [19] implicara que el diseñador tenga en cuenta este factor.

Se debe de tener en cuenta que la metodología no esta orientada a servir de guía en la búsqueda de optimización de problemas; de esta manera preguntas como: ¿dada una misión específica, cual es la configuración óptima del equipo de agentes?, no es directamente abordable desde la metodología, aunque si se espera que los pasos en el proceso ayuden al diseñador en la toma de decisiones que sean cruciales en el proyecto.

5. CONCLUSIONES Y TRABAJO FUTURO

Hay ciertos detalles de la implementación que no están cubiertos por la metodología pero que se esperan establecer como objetivos para presentaciones posteriores. Estos detalles cubren elementos como: Implementación de los dispositivos electrónicos mediante técnicas recomendadas para sistemas robóticos, Estrategias e Implementación de arquitecturas basadas en comportamientos, Aprendizaje, Estrategias para la solución de conflictos entre otros.

Debido a lo reciente del área aún queda por explorar el desarrollo de formalismos rigurosos que permitan clarificar aspectos de un sistema bajo diseño, también se debe de generalizar la terminología de manera que se llegue a un mejor acuerdo frente a conceptos como cooperación.

Propuestas posteriores explotaran más los modelos de secuencia en las descripciones de los escenarios de formación de grupo y cooperación.

Solo el desarrollo exhaustivo de proyectos con la metodología y el crecimiento del área de investigación en robótica cooperativa permitirá hacer más claras las ventajas y desventajas de trabajar las actividades aquí propuestas.

6. REFERENCIAS

- [1] Alamis R., Fleury S., Herrb M., Ingrand F., y Robert F. *Multi-Robot Cooperation in the MARTHA Project*. IEEE Robotics and Automation Magazine, Vol. 5, No. 1, 1998.
- [2] Altenburg K. *Adaptive Resource Allocation for a Multiple Mobile Robot System using Communication*. Technical Report NDSU-CSOR-TR-9404. 1994.
- [3] Collinot A., Drogoul A., Benhamou P. *Agent Oriented Design of a Soccer Robot Team*. ICMAS'96; 1996
- [4] Cao Y., Fukunaga A., Kahng A., Meng F. *Cooperative Mobile Robotics: Antecedents and Directions*. Proc. 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '95) Pittsburgh, PA, Vol. 1, pp 226-234. 1995.
- [5] Chandrasekaran B., Josephson J., Benjamins V. *Ontologies of Tasks and Methods*. IEEE Intelligent Systems, 14(1): 20-26, 1999
- [6] Donald B., Jennings J, Rus D. *Experimental Information Invariants for Cooperating Autonomous Mobile Robots*. Proceedings International Symposium on Robotics Research. 1993
- [7] DeLoach S., Matson E., Li Y.; *Applying Agent Oriented Software Engineering to Cooperative Robotics*. Proceedings of the Fifteenth International Florida Artificial Intelligence Research Society Conference. 2002.
- [8] Drogoul A, Zucker J.-D. *Methodological Issues for Designing Multi-Agent Systems with Machine Learning Techniques: Capitalizing Experiences from the RoboCup Challenge*. LIP6 – Université Paris 6. 1998
- [9] DeLoach S. *Multiagent Systems Engineering: A Methodology And Language for Designing Agent Systems*. Agent-Oriented Information Systems. 1999.
- [10] Dudek G., Jenkin M., Miliot E., Wilkes D. *A Taxonomy for Swarm Robots*. Proc. 1993 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Yokohama, Japan. 1993. pp. 441-447
- [11] B. R. Donald. *Information invariants in robotics: I. state, communication, and side-effects*. In IEEE ICRA, pages 276–283, 1993.
- [12] Evans J., Krishnamurthy B., *HelpMate, The Trackless Robotic Courier*. Lectures Notes in Control and Information Sciences. Springer Verlag. 1998.
- [13] Franklin D., Kahng A., Lewis A. *Distributed Sensing and Proving with Multiple Search Agents: Towards System-level Landmine Detection Solutions*. Proceedings of Detection Technologies for Mines and Minelike Targets, vol 2496. 1995.
- [14] Franklin S., Graesser T. *Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents*. Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages, Springer-Verlag, 1996.
- [15] Fukuda T., Sekiyama K. *Communication reduction with risk estimate for multiple robotic system*. In IEEE ICRA, pages 2864–2869, 1994.
- [16] Gerkey B., Mataric M. *Multi-robot task allocation: Analyzing the complexity and optimality of key architectures*. In Proceedings of the IEEE International Conference on Robotics and Automation, pages 3862-3867, Taipei, Taiwan, Sep 2003.
- [17] Iglesias C., Garijo M., Gonzales J., *A survey of Agent-Oriented Methodologies*; 1999; disponible en: <http://citeseer.ist.psu.edu/iglesias99survey.html>
- [18] Iglesias C., Garijo M., Gonzales J., Velásquez J. *Analysis and Design of Multiagent Systems using MAS-CommonKADS*. Disponible en: <http://citeseer.ist.psu.edu/iglesias98analysis.html>
- [19] Jones J., Seiger B., Flynn A. *Mobile Robots. Inspiration to Implementation*. Second Edition. A K Peters Natik, Massachusetts. 1998
- [20] Khosla, P., Navarro, L., Paredis, Ch. *A Beacon System for the Localization of Distributed Robotic Teams*. Carnegie Mellon University. 1999.
- [21] Kendall E. *Agent Roles and Role Models*. Intelligent Agents for Information and Process Management (AIP'98). 1998
- [22] K. VanLehn, editor. *Architectures for Intelligence: the 22nd Carnegie Mellon Symposium on Cognition*. Laerence Erlbaum Associates. 1991
- [23] Mäkelä, H. *Outdoor Navigation of Mobile Robots*. Tesis Doctoral. University of Technology. 2001

- [24] Müller J.P. *The Design of Intelligent Agents. A Layered Approach. Lecture Notes in Artificial Intelligence, Vol. 1177*. Ed. Springer-Verlag. 1996.
- [25] Mataric M. Issues and Approaches in Design of Collective Autonomous Agents. 1994
- [26] Mataric M. *Behavior-Based Control: Examples from Navigation, Learning, and Group Behavior*. 1997. Disponible en: <http://citeseer.ist.psu.edu/mataric97behaviorbased.html>
- [27] Nilsson N. *Inteligencia Artificial. Una nueva Sintesis*. McGraw Hill. 2001
- [28] Odell J., Parunak H., Bauer B. *Extending UML for Agent*. 2000. Disponible en: <http://www.jamesodell.com/ExtendingUML.pdf>
- [29] Parker L. *Toward the automated synthesis of cooperative mobile robot teams*. 1998. Disponible en: <http://citeseer.ist.psu.edu/parker98toward.html>
- [30] Rekeleitis I., Dudeck G., Milios E. *MultiRobot Exploration of a Unknown Environment, Efficiently Reducing the Odometry Error*. IEEE Proceeding of International Joint Conference in Artificial Intelligence, vol 2. 1997.
- [31] Rekeleitis I., Sim R., Dudeck G., Milios E. *Collaborative Exploration for the Construction of Visual Maps*. 2001. Disponible en: <http://citeseer.ist.psu.edu/449413.html>
- [32] Sandt F., Panpagnin L. Perception for a transport robot in Public Environment. Proceeding of IROS. 1997.
- [33] Tambe M. *Towards Flexible Teamwork*. Journal Artificial Intelligence Research. 1997.
- [34] Veloso M., Nardi D. *Special Issue on Multirobot Systems*. Proceedings of the IEEE, Vol. 94, No. 7, July 2006
- [35] Vernon C. A Principled Design Methodology for Minimalist Multi-Robot System Controllers. Ph.D. Thesis. UNIVERSITY OF SOUTHERN CALIFORNIA. 2005
- [36] Wooldridge M., Jennings N. R., Kinny D.. *A Methodology for Agent-Oriented Analysis and Design*. 1999. Disponible en: <http://www.csc.liv.ac.uk/~mjw/pubs/agents99.pdf>