

Modelo de responsabilidad compartida con sistemas multiagentes industriales para la resolución de problemas cooperativos

Juan José Mora Flórez <jjmora@silver.udg.es>

*Grup d'Enginyeria de Control i Sistemes Intel·ligents (eXiT)
Institut d'Informàtica i Aplicacions (IiiA)*

*Universitat de Girona (UdG) & European Associated Laboratory-Intelligent Systems and Advanced Control (LEA-SICA)
An Luis Santaló s/n, 17071 Girona (Spain)*

Resumen — Este artículo presenta un modelo denominado de Responsabilidad Compartida, utilizado para la resolución conjunta de problemas con tecnología de inteligencia artificial distribuida (DAI). El modelo se fundamenta en intenciones compartidas como su núcleo central y establece básicamente las precondiciones necesarias antes que la cooperación inicie, así como el comportamiento de los individuos involucrados en la resolución conjunta del problema, bajo condiciones normales y anormales de la actividad.

Este modelo teórico se fue usado para desarrollar la implementación de una estructura de cooperación de propósito general. Sus beneficios cualitativos y cuantitativos se evaluaron en el escenario real del manejo de transporte de energía.

Palabras claves – agentes, inteligencia artificial, intención compartida, modelo de cooperación.

1. INTRODUCCIÓN

El software ha venido reemplazando muchos de los elementos análogos que en antaño se encontraban en las fábricas y en las grandes instalaciones industriales. Se ha constatado así mismo que en la medida que el desarrollo computacional se hace más sofisticado, incrementa su tamaño y complejidad, generando serias limitaciones en su desempeño y mantenibilidad.

A partir de lo expuesto, se han generado actividades de reingeniería que consideran los paradigmas existentes. A partir de éstas han surgido básicamente cuatro teorías o corrientes para el desarrollo de métodos de solución de problemas: la primera propone una basta recopilación de una gran cantidad de conocimiento lógico o de sentido común, otros por el contrario proponen un modelo con librerías compartibles y reutilizables de resolución conjunta de problemas, un tercer grupo sugiere el desarrollo de sistemas generalizados para abordar todo tipo de problemas como vía mas adecuada, mientras que un cuarto y último grupo argumenta que la mejor solución es el desarrollo de componentes pequeños y manejables que se puedan comunicar y cooperar.

En sistemas de inteligencia artificial distribuida (DAI), los agentes involucrados en la solución de los problemas, cooperan para alcanzar sus metas locales y las metas globales de la comunidad. El trabajo conjunto es muy importante debido a la dependencia entre las acciones de los agentes, la necesidad de cumplir con restricciones globales y debido que ninguno de los individuos es lo suficientemente competente, ya que no tiene los recursos o la información suficientes para resolver el problema por sí solo.

En este tipo de modelos existe gran interdependencia porque la mayoría de las actividades desarrolladas por cada uno de los agentes tienen un impacto en la comunidad. De la misma manera, estos modelos de agentes requieren que cada uno optimice su labor pero que en conjunto satisfagan las restricciones globales.

Finalmente aparece gran cantidad de problemas que deben ser abordados en un entorno de cooperación, de tal forma que diferentes agentes tienen diferente experiencia, diferentes recursos e información, la cual debe ser combinada para la satisfacción de la necesidad planteada.

En el problema aquí presentado, se utiliza un modelo de DAI como medio para hacer frente a la complejidad industrial por tres razones básicas: a) Divide y vencerás, entre más pequeños son los problemas, menos complejos son, b) La naturaleza distribuida de la aplicación encaja perfectamente con la naturaleza distribuida del problema y c) Muchas de las organizaciones tienen software preexistente que puede utilizarse de forma integrada en una aplicación.

Uno de los principales problemas consiste en asegurar que el sistema funcione de forma coordinada y coherente cuando ocurran acontecimientos no planeados. Algunos de los posibles problemas pueden ser: a) nueva información se encuentra disponible, lo cual hace parecer que el procesamiento llevado a cabo es obsoleto, b) se reciben requisiciones inesperadas, o los puntos de sincronización inter-agentes no son respetados, c) no se reciben respuestas a requisiciones, entre otras. Estas incoherencias ocurren principalmente por el hecho que los agentes participantes no tienen el conocimiento suficiente del proceso de resolución conjunta de problemas para operar en dominios inciertos, dinámicos y complejos.

Como solución propuesta para estos problemas se presenta un modelo que no solo prescribe como se deben comportar los agentes cuando todo progresa de forma planeada, sino también cuando algo no resulta como lo estipulado. Este modelo se denomina Responsabilidad Compartida "*Joint Responsibility*"

Con este modelo se desarrolló una versión extendida de GRATE (Generic Rules and Agent model Tested

Environment) denominado *GRATE**, cuya coherencia fue probada en medios de hostilidad variada.

2. EXPERIENCIAS INICIALES CON SISTEMAS MULTI AGENTES INDUSTRIALES

Si los agentes tuvieran una capacidad infinita de procesamiento y el conocimiento completo de las metas, las creencias, las acciones y las interacciones de toda la comunidad, sería posible conocer que está haciendo cada uno de ellos en cada instante de tiempo, así como lo que intentaría hacer en el futuro. Bajo estas condiciones ideales, los sistemas serían perfectamente coordinables y su costo relativamente bajo. La realidad, es que en las aplicaciones industriales encontramos algunos inconvenientes como a) Limitaciones de ancho de banda que hace imposible estar continuamente informado de la evolución del sistema, b) Los agentes no pueden razonar continuamente acerca de todas las actividades que ocurren en su comunidad y mantener simultáneamente su procesamiento local.

Una de las aplicaciones industriales más populares es aquella en la cual se dota a un agente con un componente con una visión completa del sistema y con un número de subcomponentes funcionalmente distribuidos. Estos sistemas generalmente están ordenados de forma jerárquica y tienen vínculos claros de comunicación. Estos componentes aumentan la mantenibilidad pero requiere que todo el control y la coordinación inter-componentes esté centralizada, lo cual hace que a medida que el componente se vuelve más complejo, el cuello de botella se vuelve más crítico. Tiene la desventaja adicional que si el controlador general falla, el sistema se vuelve completamente inútil.

Se plantea entonces un modelo descentralizado tanto de control como de datos, el cual permite aliviar el cuello de botella de las actividades de control e incrementa la flexibilidad de la coordinación entre subcomponentes. Este nuevo modelo genera agentes con autonomía para la realización de nuevas tareas pero es más difícil de alcanzar un

comportamiento coherente global. Esta dificultad es especialmente crítica donde las percepciones y las acciones pueden fallar y cuando el entorno evoluciona dinámicamente, como es el caso de las aplicaciones industriales típicas.

Para dar una base sólida a la discusión que posteriormente se desarrollará, se presenta el escenario de trabajo, que es tomado de una aplicación de manejo del transporte de la energía eléctrica. El transporte de energía eléctrica está precedido por la fase de generación y seguido por la fase de subtransmisión regional y luego la de distribución local. El manejo del transporte de la energía involucra muchas tareas separadas pero interrelacionadas tales como la distinción entre disturbios de red y actividades de mantenimiento preplaneadas, identificación del tipo, origen y la extensión de la falla, la determinación de la mejor manera de restablecer la red de tal forma que la energía de racionamiento sea mínima.

Aparece a partir de este problema una gran variedad de actividades las cuales involucran un número considerable de agentes. Para propósitos de entender la metodología propuesta se consideraran únicamente tres agentes y el problema de detección y diagnóstico de fallas. Los agentes para análisis de alarmas (Alarm Analysis Agent - AAA) y el agente para identificación del área afectada (Blockout Area Identifier - BAI), desarrollan el diagnóstico, mientras que un tercer agente de interfase del sistema de control (Control System Interfase - CSI) detecta el disturbio inicialmente y luego monitoriza la evolución de la red. Este escenario se seleccionó debido a que es típico de una clase de comportamiento denominado funcionalmente cooperación exacta, en la cual los agentes no poseen toda la información necesaria para resolver su problema completa y exactamente, lo cual es típico de los sistemas industriales.

Inicialmente cada AAA y BAI contaban con un CSI propio con el cual comparaban mensajes pero no podían detectar disturbios. De esta forma se mantenían a dos sistemas de control CSI continuamente invocados por rutinas como mantenimiento, y estos adicionalmente tampoco

podían intercambiar información de carácter predictivo. Como solución se plantea utilizar una aproximación multiagente para desarrollar un sistema común y más sofisticado de control CSI, que permita a los agentes hacer un diagnóstico cooperativo.

Con la aproximación planteada se mejora el comportamiento, pero si ocurre un suceso no planeado, entonces el sistema no funciona muy bien. Estos problemas se presentan generalmente por: a) Muchas de las acciones no fueron explícitamente representadas, b) aún cuando la acción social fuera explícitamente representada, el modelo pierde vigencia por la miles de asunciones cruciales no mencionadas que ello involucra.

2.1 Acciones conjuntas no explícitas

En este tipo de aplicaciones, la representación que cada uno de los agentes tiene acerca del otro controla el intercambio de información a nivel de dominio. Estos modelos de agentes contienen información acerca de las capacidades, metas, procedimientos, e intereses de los otros miembros de la comunidad y están instanciados por el diseñador cuando construye el sistema multiagente.

En el escenario de diagnóstico de fallas, el CSI representa el hecho que el AAA y el BAI están interesados en la información de que un disturbio ha ocurrido, y la representación que tiene AAA de BAI indica que éste es capaz de identificar aproximadamente la localización de la falla.

Cuando CSI detecta un disturbio no planeado, informa a AAA y a BAI y entra entonces en el modo de monitorización de red. CSI no tiene en cuenta el efecto que su información causa en los otros miembros de la comunidad y únicamente cree que ellos tienen el interés en conocer que un disturbio ha sido detectado. Luego de recibir la información AAA y BAI inician el diagnóstico de la falla. AAA hace un análisis aproximado y rápido de un gran número de soluciones potenciales y luego hace una validación un poco más lenta. Cuando tiene la

información acerca del área afectada de parte de BAI, ésta es usada para seleccionar un número de opciones factibles en la fase de validación.

De esta forma AAA, BAI y CSI están claramente involucrados en una tarea conjunta y sus actividades locales contribuyen a alcanzar una meta global que consiste en localizar la falla. Sin embargo, la revisión de la estructura de los agentes revela que ellos están trabajando en actividades locales con medios locales, debido que así lo ha especificado el diseñador. Entonces esta no es una representación del hecho que cada tarea es parte de una más grande que implica una actividad de cooperación. Considerando por ejemplo que AAA decida abandonar el diagnóstico, este agente no sabe que CSI y BAI deben ser informados para que ellos detengan sus actividades.

Esta representación implícita del conocimiento es suficiente cuando todo se procesa de acuerdo a un plan y nada falla dentro del mismo. Para evitar la pérdida de la representación de una acción conjunta, se debe colocar más información en los lugares de interés de los modelos de los agentes, por ejemplo las representaciones que AAA tiene de CSI y BAI se debe extender a un estado en que estos últimos estén interesados en informarse cuando AAA desiste de sus acciones. Una desventaja de este tipo de modelos es que el diseñador debe identificar y conocer todos los eventos que puedan causar la falla de la actividad conjunta, lo cual es una tarea muy difícil de acometer.

2.2 Modelo de cooperación no definido claramente

El proceso de cooperación también puede ocurrir a través del voluntariado de la información relevante, por lo cual GRATE provee también la facilidad de establecer acciones sociales explícitas. Por ejemplo para AAA se debe especificar que debe recibir información acerca del área de falla. Después de examinar los modelos de los agentes, AAA entiende que únicamente puede saber esto por una requisición específica de BAI, ya que el AAA u otros agentes no tienen una salida que de como resultado el área afectada por la falla. Si se asume que BAI acepta el

requerimiento, se establece entonces una acción social específica.

Aún con estas acciones sociales explícitas, los agentes GRATE usan un modelo implícito de cooperación para guiar las acciones e interacciones. Como ejemplo se tienen las reglas mostradas en la figura 1, las cuales forman parte de la valoración de la funcionalidad propuesta.

R1: if successfully finish recipe R and
 R was performed because of a request by agent A
 then inform A that R has finished and supply it with
 the results which were produced

R2: if recipe R has an agreed deadline D and
 R will not be completed by D and
 R was performed because of a request by agent A
 then inform A that R will not be completed by D

Figura 1: Reglas del modulo de valoración de la situación de GRATE

La regla 1 especifica que cuando hay un requerimiento específico de cooperación que ha sido completado, el agente que desarrolló el procesamiento de la tarea debe informar al que la requirió, que el resultado está listo. La regla 2 representa el principio que cuando se requiere un servicio este tiene un plazo para cumplirse y usualmente significa que este debe ser sincronizado con otras de las actividades del originador.

Como conclusión se tiene que a) BAI y CSI deben ser informados que AAA abandonó su procedimiento de diagnóstico. Esto es de fundamental importancia debido que las actividades de los dos primeros son significativas únicamente en el contexto de las acciones conjuntas y esto nunca ocurre si AAA no finaliza su labor, b) CSI necesita informar a AAA y BAI de su identificación errónea de una falla transitoria, porque la motivación de un acción conjunta no se mantiene y todos los que conforman el equipo de trabajo cesan su procesamiento asociado.

R1 y R2 muestran características de reglas de sistemas expertos de la primera generación que tienen el conocimiento de superficie asociativo, el cual es suficiente para el efecto de inferencia, pero no representa el dominio fundamental. En este caso, el dominio fundamental es el proceso de solución de

problemas cooperativos, antes que el diagnóstico de falla o la restauración de la red. En GRATE no hay un modelo explícito de cómo cooperar, las reglas representan una asociación directa entre un estado de resolución del problema de un agente y sus acciones.

Una mejor solución debe proveer a los agentes con una representación explícita del modelo de cooperación que se va a emplear. Este modelo debe especificar: a) Como se deben comportar los agentes cuando realizan sus actividades locales respecto a la acción conjunta, b) Como debe iniciar la acción, c) Como se deben resolver los problemas con la acción conjunta, d) Como deben actuar los miembros del equipo cuando aparecen los problemas y e) como deben los agentes organizar sus actividades locales en respuesta a los problemas con la acción conjunta.

Este método es consistente y constituye una buena forma para hacer frente a los problemas, sin embargo los agentes GRATE no poseen los procedimientos necesarios que los habiliten para operar en ambientes inciertos y por esta razón su nivel de desempeño no es muy alto.

3. MODELOS EXPLICITOS DE SOLUCIÓN DE PROBLEMAS COOPERATIVOS

Un modelo de cooperación requiere dos tipos de construcciones básicas: a) la asociada a la definición del comportamiento individual en un contexto social y b) el asociado al comportamiento social por si mismo.

De acuerdo a una gran cantidad de investigadores, se plantea que la definición del comportamiento individual en un contexto social debe ser controlado por el concepto central de las intenciones. Bratman, presenta tres facetas importantes del comportamiento individual gobernado por las intenciones: a) cuando un agente decide alcanzar una meta, éste realiza algunas actividades para alcanzar ese estado de eventos, b) Las intenciones son usadas para organizar acciones futuras y c) las intenciones plantean problemas por medio del análisis final, lo cual ocurre porque los agentes frecuentemente acometen ellos

mismos a una meta, antes de haber trabajado en los detalles de cómo ésta será alcanzada.

Las propiedades que deben tener las intenciones para cumplir con estos tres roles son: a) Deben ser internamente consistentes, así como también consistentes con sus creencias y b) deben tener un grado de estabilidad, pero estas no deben ser completamente inflexibles

Las intenciones deben cumplir un criterio específico de un modelo de cooperación explícito. Éstas definen un comportamiento local e individual, cuando todo va progresando satisfactoriamente; a través de la definición de convenciones ofrecen un entendimiento de los tipos de dificultades que pueden resultar durante la solución del problema.

Dado el éxito y la prestancia del modelamiento individual usando intenciones, las intenciones conjuntas son consideradas como los medios naturales para describir inherentemente el fenómeno de cooperación. De acuerdo con Jennings, los siguientes aspectos deben ser considerados como importantes en la resolución conjunta de problemas: a) los agentes deben tener una meta conjunta, b) los agentes deben estar de acuerdo en que ellos desean colaborar para alcanzar su meta conjunta, c) los agentes deben estar de acuerdo en un procedimiento común para alcanzar la meta conjunta, d) las acciones desarrolladas por diferentes agentes en el contexto de las acciones conjuntas deben ser interdependientes y e) los agentes deben tener convenciones para monitorizar la viabilidad de sus compromisos.

Casi todas las formulaciones sostienen el hecho de que un acción conjunta requiere de una meta común, con la cual el grupo muestra su deseo por alcanzarla de una forma cooperativa. El propósito de una meta común es el de proveer un medio de unión de las acciones individuales en un todo cohesivo, lo cual permite establecer la diferencia entre competencia y cooperación.

Se debe tener presente sin embargo que no siempre la existencia de una meta común asegura la solución cooperativa del problema. Se necesita entonces una

fórmula común que provea un contexto para el rendimiento de las acciones de la misma forma que las metas globales guían los objetivos de los individuos. El procedimiento común provee de un contexto para el desarrollo de las acciones de la misma manera que las metas conjuntas guían los objetivos de los individuos. Específicamente, si se incluye el requerimiento de procedimiento común en la definición, se coloca un rol funcional adicional en las intenciones conjuntas. Esto guía a los participantes hacia un acuerdo de un curso común de acciones.

Para que una intención conjunta esté activa, debe existir un procedimiento aceptado por todos y bajo el cual trabajan los agentes, lo cual refleja el hecho que los miembros del equipo deben creer que eventualmente ellos deben estar de acuerdo sobre, y trabajar bajo un procedimiento común con respecto a su meta conjunta.

Un aspecto importante del modelo de cooperación deseado es que éste especifica el criterio contra el cual las intenciones conjuntas deben ser monitorizadas, así como el comportamiento que se debe adoptar cuando ésta funciona mal.

Dado que ninguna de las formulaciones existentes cumple con este modelo se ha llegado a la conclusión que se debe plantear un nuevo modelo de intenciones conjuntas, el cual cumpla con los requerimientos especificados.

3.1 Modelo con responsabilidad conjunta

La responsabilidad compartida se especifica formalmente usando lógica modal y temporal. Como muchos otros modelos, el modelo de responsabilidad se puede modelar con dos niveles diferentes, un bajo nivel en el cual se definen nociones como metas, procedimientos, interdependencias, acciones, entre otras y un alto nivel de conceptos tales como compromiso conjunto para una meta conjunta y compromisos conjunto para un procedimiento común.

El objeto de este artículo es mostrar como las intenciones del modelo formal pueden ser usadas para

guiar el proceso de construcción de comunidades cooperativas robustas.

3.1.1. Metas conjuntas

El modelo de responsabilidad conjunta usa un modelo en el cual el compromiso está basado en la noción de metas persistentes conjuntas, las cuales a su vez están basadas en el concepto de metas alcanzadas. El concepto de Metas alcanzadas define el estado mental de los individuos que participan en un equipo, los cuales trabajan hacia una meta conjunta con una motivación específica.

Un agente α tiene una *meta exitosa débil* relativa a su motivación q , para causar p sí y solo sí cualquiera de lo siguiente es verdadero: a) α no cree que p es verdadero y que p puede ser eventualmente verdadero como meta, b) α cree que p es verdadero, puede que nunca sea verdadero o es irrelevante, pero tiene una meta de hacer el estado de p mutuamente creíble por todos los miembros del equipo.

Esa *meta exitosa débil* tiene cuatro casos separados: a) α tiene a p como meta normal exitosa, b) α piensa que p es verdadera y quiere que este hecho sea conocido por todos, c) α cree que p nunca puede ser verdadero y quiere que todos conozcan este hecho, d) α cree que la motivación de p no es completamente valida y quiere que todos los agentes tengan precaución al respecto de este hecho.

Un equipo de agentes tiene una *meta conjunta persistente* relativa a q , para alcanzar p sí y solo sí:

- a) Todos mutuamente creen que p es normalmente falso.
- b) Todos creen que todos desean que p sea eventualmente verdadero, y hasta que ello ocurra por la mutua creencia puede ser que p es verdadero, ó p nunca puede ser verdadero ó q es falso. Ellos pueden continuar mutuamente creyendo que cada uno de ellos tiene a p ; como una *meta exitosa débil* relativa a q .

Si un grupo de agentes tienen un compromiso conjunto para alcanzar p , ellos inicialmente piensan

que en efecto la meta es **p**, posteriormente algunos no confían que **p** pueda ser la meta, por lo cual existe una creencia débil hacia este compromiso. Esto último puede ocurrir debido que un grupo de agentes descubrió que **p** es imposible o irrelevante. Sin embargo una creencia débil sobre la meta puede persistir aunque todos estén informados de la pérdida del compromiso. Lo anterior significa que los agentes pueden confiar en el compromiso de los otros, primeramente por el objetivo global (Meta exitosa normal) y luego si es necesario por la creencia mutua del estado del objetivo (Meta exitosa débil). La existencia de una meta exitosa débil significa que pueden detectar el problema y deben esforzarse por informar a todos aquellos que no lo han hecho.

Lo anterior muestra como este modelo de cooperación para las metas persistentes conjuntas tiene las siguientes características: a) Especifica que los agentes deben mantener sus compromisos con respecto a la meta conjunta, b) Presenta la convención por la cual los agentes deben monitorizar su compromiso con la meta conjunta y c) describe como se deben comportar los agentes localmente y con respecto a los otros, cuando los compromisos han sido revocados.

3.1.2. *Procedimientos comunes*

El compromiso común para el procedimiento acordado comúnmente se especifica a través de la noción del compromiso de procedimiento conjunto, el cual a su turno se basa sobre el concepto de compromiso de procedimiento individual.

El compromiso de procedimiento individual especifica que cada uno de los miembros del equipo debe permanecer comprometido y desarrollar las acciones con las que esté de acuerdo con realizar, dentro del contexto del procedimiento común, a menos que una de las siguientes circunstancias aparezcan:

- a) La salida deseada ya se encuentra disponible.
- b) El procedimiento acordado no tiene el resultado deseado.
- c) Una de las acciones especificadas no puede desarrollarse.

- d) Una de las acciones aceptadas no puede desarrollarse adecuadamente.

Las anteriores situaciones definen entonces cuando el agente puede detectar por si mismo los problemas relacionados con un procedimiento común. Esto se logra típicamente porque el agente que está involucrado en la acción de falla, puede observar directamente el comportamiento de su comunidad o porque alguien le informa. Esta última se denomina detección indirecta y es importante por la naturaleza cooperativa. Si un agente no coopera, el proceso puede fallar. Un agente puede perder su compromiso por dos causas fundamentales a) detecta un problema o b) Es informado de un problema por uno de los miembros del equipo.

El compromiso de procedimiento individual define exactamente como se debe comportar el agente con respecto a las acciones de resolución de problemas dentro del contexto de las acciones conjuntas. Éste también define la convención con la cual los agentes deben monitorizar sus compromisos con los procedimientos comunes. A partir de aquí, el compromiso de procedimiento conjunto especifica que cuando un agente pierde su compromiso con el procedimiento común, éste debe informar a los demás miembros de la comunidad. Como consecuencia de lo anterior, el equipo de trabajo puede reconsiderar todos sus procedimientos y tomar la decisión de abandonarlos o redefinirlos.

3.1.3. *Responsabilidad conjunta*

Responsabilidad conjunta tal como se ve en la figura 2, provee un modelo explícito de cooperación el cual cumple con todo lo propuesto en las anteriores secciones. Esto requiere que todos los miembros del equipo $\alpha_1 \dots \alpha_n$ tengan una meta persistente para lograr la meta conjunta σ y que todos ejecuten el procedimiento común Σ , de acuerdo con los principios del compromiso con el procedimiento común. Además todos los agentes mutuamente conocen que están haciendo los otros mientras desarrollan su papel.

```

JOINT-RESPONSIBILITY ({ $\alpha_1 \dots \alpha_n$ },  $\sigma, \Sigma$ ) ...
  FORALL  $\alpha_i \in \{\alpha_1 \dots \alpha_n\}$ 
    WHILE Normal-Achievement-Goal( $\alpha_i, \sigma$ ) AND Individual-Recipe-
      Commitment( $\alpha_i, \sigma, \Sigma$ ) DO
      PARALLEL
        Honour commitments to joint goal and common recipe
        Monitor rationality of commitment to  $\sigma$ 
        Monitor rationality of commitment to  $\Sigma$ 
      END-PARALLEL
      Suspend processing of local activities related to  $\Sigma$ 
      CASE Reason for non commitment OF
        NOT Normal-Achievement-Goal( $\alpha_i, \sigma$ ): Abandon commitments to
           $\sigma$  and subsequent actions in  $\Sigma$ 
        NOT Individual-Recipe-Commitment( $\alpha_i, \sigma, \Sigma$ ): IF remedial action
          available
          THEN select possible remedies
          ELSE seek assistance to devise new common recipe
      END-CASE
  END-FORALL

```

Figura 2: Modelo de responsabilidad conjunta de la resolución de problemas cooperativos

Se necesitan dos tipos de compromiso conjunto que de acuerdo a los diferentes estados de la acción desarrollada. Cuando el compromiso por la acción se pierde, la acción conjunta se termina; si la meta se alcanza, el grupo completa satisfactoriamente su objetivo. Si la motivación no está presente o la meta nunca será alcanzada, no hay punto de continuación. Sin embargo si el grupo pierde su compromiso con el procedimiento común, este puede ser un procesamiento útil para ser desarrollado. Por ejemplo si un procedimiento se convierte en invalido, los agentes pueden intentar diferentes secuencias de acciones las cuales producen el mismo resultado. Similarmente, si una acción de un procedimiento no se desarrolla correctamente, los agentes pueden reprogramar sus actividades.

4. IMPLEMENTACIÓN DE AGENTES DE RESPONSABILIDAD CONJUNTA

El modelo de responsabilidad sugiere un alto nivel de arquitectura para la cooperación de agentes, en la cual las intenciones conjuntas juegan un papel importante en coordinar las acciones futuras y controlar la ejecución de las acciones actuales. La adopción del modelo de responsabilidad sin embargo tiene algunas restricciones, consistentes en que el agente no es neutral en término de implementación. Sin embargo las descripciones son convenientemente de un alto nivel para asegurar que hay suficiente libertad para implementación de los conceptos planteados. Por

ejemplo el modelo no especifica como son representadas las intenciones, qué mecanismos son utilizados para obtener los acuerdos, como tampoco como desarrollar el procedimiento común.

El concepto de responsabilidad se debe implementar en un entorno computacional para controlar aplicaciones industriales y se utilizará un aproximación basada en reglas, dado su menor coste en recurso computacional y tiempo de ejecución. GRATE ya provee la semántica para el modelo de Responsabilidad Conjunta. Las reglas usadas se dividen en dos grupos: a) las que aseguran la cooperación del grupo y b) las que desarrollan las funciones cooperativas por sí mismas.

Las reglas del literal a) o de aseguramiento de la situación, son necesarias por: deciden cuando una cooperación en una meta conjunta es apropiada, desarrollan y obtienen acuerdos en procedimientos comunes, aseguran y mantienen los compromisos existentes, aseguran que los nuevos compromisos son consistentes con los existentes, monitorizan el estudio de resolución de problemas con respecto a la pérdida de compromiso y deciden que acciones se deben tomar si se pierde el compromiso. Las reglas que controlan las interacciones cooperativas deben asegurar que todos los miembros del equipo estén informados si uno de los agentes desiste de su compromiso, en tales circunstancias los demás agentes tienen que estar atentos a esa eventualidad y a la acción remedial propuesta. El modelo formal prescribe las estructuras de datos que incluyen: Metas, procedimientos, intenciones e intenciones conjuntas.

Existen algunas suposiciones de simplificación propuestas para facilitar una implementación razonable del modelo así: a) La comunicación no es perfecta y los agentes conocen el retraso de tiempo del mensaje, b) aunque el conocimiento mutuo tiene muchas propiedades teóricas atractivas, este concepto no es fácil de implementar en aplicaciones prácticas, por lo cual se necesita una aproximación computacional, que en GRATE* puede llevar a creencias con cierto nivel de anidamiento (Todo el mundo cree que todo el mundo cree), c) Para

desarrollar la coordinación de actividades los agentes comparten una referencia de tiempo de un reloj global, y d) los agentes pueden predecir con un razonable grado de exactitud el tiempo empleado en cada una de sus actividades.

4.1 Arquitectura de un agente GRATE*

Los agentes GRATE* tienen dos componentes identificables: a) capa de cooperación y control y b) el sistema al nivel de dominio tal como se muestra en la figura 3. El sistema al nivel de dominio resuelve los problemas de organización, los cuales se expresan como tareas. La capa de cooperación es un controlador metanivel el cual opera en el sistema en el nivel de dominio y asegura que las actividades de cada agente estén coordinadas con los demás.

La capa de cooperación tiene tres módulos de solución de problemas principales: módulo de cooperación, módulo de aseguramiento de la situación y módulo de control. Cada módulo es implementado en forma separada como un sistema de producción “*forward-chaining*” con su propio mecanismo de inferencia y su memoria de trabajo local.

El módulo de control es la interface para el sistema al nivel de dominio y es el responsable por el manejo de todas las interacciones con el mismo.

El módulo de aseguramiento de la situación toma las decisiones que afectan los otros dos módulos, decide cuales actividades se desarrollan de forma local y cuáles se delegan, cuáles de las requisiciones de cooperación se mantienen, como deben ser realizadas las requisiciones y que acciones se deben tomar como resultado del ingreso de nueva información.

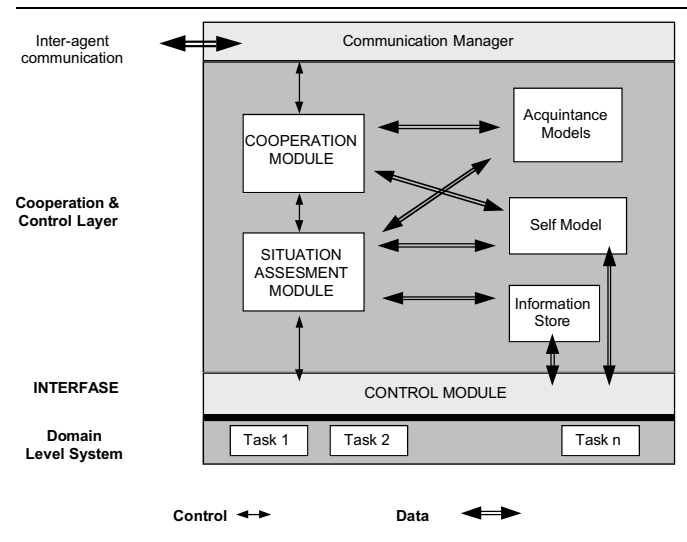


Figura 3: Arquitectura del agente GRATE*

El módulo de cooperación es el responsable del manejo de las actividades sociales de los agentes y soporta tres objetivos asociados a este rol: a) Establecer nuevas interacciones sociales (Encontrar un agente que pueda enviar parte de la información deseada) b) Monitorizar la actividad cooperativa y c) Responder las requisiciones cooperativas de otros agentes.

Los demás componentes proveen funciones de soporte. El componente de almacenamiento de información guarda todos los datos que ha generado el sistema al nivel de dominio o aquellos que han sido recibidos como resultado de la interacción con otros agentes. El componente del modelo de conocimiento o familiaridad corresponde a las representaciones de los otros agentes y el componente del modelo propio es una representación del sistema al nivel de dominio. El encargado de enviar y recibir los mensajes a los otros agentes de la comunidad, se conoce como administrador de comunicación.

4.2 Desarrollo de condiciones adecuadas para la cooperación

Antes que la cooperación pueda desarrollarse, se debe establecer una acción conjunta entre el grupo de agentes. La necesidad para, o el deseo de, todas las nuevas actividades son detectadas por el módulo de aseguramiento de la situación de los agentes. Luego que un agente ha seleccionado el procedimiento

adecuado para alcanzar la meta deseada, se determina si debe completar la actividad localmente o si debe buscar ayuda con los otros miembros de la comunidad. Esta decisión se basa en el número de componentes del procedimiento que el agente tiene que desarrollar. Hay tres salidas potenciales de este análisis: a) La acción conjunta definitivamente se necesita, b) la acción debe ser resuelta localmente y, c) se necesita alguna asistencia pero no lo suficiente para garantizar la acción conjunta en su totalidad.

Si toda la acción conjunta está completamente garantizada, el agente que detecta la necesidad se convierte en el organizador. El papel del organizador consiste en: a) Identificar los miembros de la comunidad que quieren participar en la acción conjunta, b) Influenciar a los miembros del equipo para que estén de acuerdo con el procedimiento común y también con el modelo de cooperación de responsabilidad y c) Diseñar el procedimiento común. Es importante anotar que en aplicaciones industriales antes que tener un protocolo muy extenso en el cual cada prerequisite se cumpla de forma separada, se tiene una versión concisa en la cual varias de las condiciones se encuentran en uno solo de los mensajes intercambiados.

De acuerdo con el protocolo mostrado anteriormente, se presenta que GRATE* tiene dos fases tal como se muestra en la figura 4. La primera fase con los agentes que quieren participar y las bases fundamentales y la segunda especifica quien y en que instante debe desarrollar cada acción. Como el organizador no tiene una representación completa de todas las entidades dentro de la comunidad, éste no se preocupa por la existencia de compromisos o deseos de todos sus miembros potenciales. Por lo tanto, se deben alcanzar tiempos exactos a través de acuerdos mutuos en lugar de que sean fijados por el organizador.

PHASE 1

Organiser detects need for joint action to achieve goal G and determines that recipe R is the best means of attaining it
Organiser contacts all acquaintances capable of contributing to R to determine if they will participate in the joint action using the Responsibility cooperation model
Let: Ω = set of willing acquaintances

PHASE 2

FORALL actions in R
select agent A $\in \Omega$ to carry out action $\theta \in R$ (criteria: minimize number group members) calculate time (t_θ) for θ to be performed based on temporal orderings of R and the anticipated communication delay
send (θ, t_θ) proposal to A
A evaluates proposal against existing commitments (C's):
IF no-conflicts (θ, t_θ) THEN create commitment C θ for A to (θ, t_θ)
IF conflicts(θ, t_θ), C) ^ priority(θ) > priority(C)
THEN create commitment C θ for A to (θ, t_θ) and reschedule C
IF conflicts(θ, t_θ), C) ^ priority(θ) < priority(C)
THEN find free time ($t_\theta + \Delta t_\theta$), note commitment C θ and return updated time to leader
Return acceptance or modified time to team organiser
IF time proposal modified THEN update remaining action times by Δt_θ
END-FORALL

Figura 4: Protocolo de planeación distribuida de GRATE*

Ahora, y considerando nuevamente el escenario del sector eléctrico de transmisión, se reconsideraran los conceptos mencionados anteriormente. Después del deseo de establecer la acción conjunta, el organizador (AAA) inicia una representación de la intención conjunta en su propio modelo.

La parte de motivación (**motivation**) indica que la razón para desarrollar una acción conjunta consiste en que se ha sido detectado un disturbio.

La parte de procedimiento (**recipe**) consiste de una serie de acciones que se deben desarrollar teniendo en cuenta las restricciones, para producir la salida deseada. Este procedimiento utiliza la filosofía de representación explícita de las acciones conjuntas y de cómo combinarlas para alcanzar la meta global. El procedimiento indica lo que se debe hacer, no quien o cuando lo debe desarrollar. Los tiempos detallados están en la segunda fase del protocolo.

La parte de prioridad (**priority**) indica la importancia de la intención y es usada como la base computacional de deseo de ejecutar la acción. El valor más alto corresponde a la acción que más se desea ejecutar. Las prioridades son una combinación de dos componentes: un valor intrínseco (estático) asociado con cada procedimiento y un componente dinámico el cual provee la flexibilidad necesaria para referir el problema al contexto de solución.

El componente correspondiente al estado (**status**) se refiere a la fase actual del protocolo de

planeamiento y puede tener los siguientes estados: Establecimiento del grupo, desarrollo de la solución o ejecución de la acción cooperativa.

La parte de la salida (**outcome**) está relacionada con los resultados esperados al desarrollar la acción en curso.

Los participantes indican la estructura organizacional del grupo y el estado actual del desenvolvimiento de cada agente.

La parte de (**bindings**) es usada exclusivamente en la segunda fase del protocolo.

La contribución propuesta (**Proposed-contribution**) guarda estos agentes, los cuales son adjudicados por el organizador para ser capaces de contribuir con el acto conjunto y también especificar cuando ellos están de acuerdo para hacer una contribución.

Luego de identificar la necesidad de acción conjunta, se puede iniciar el proceso de cooperación. La primera fase del protocolo determina cuales de los agentes conocidos están dispuestos a participar.

El proceso de establecimiento de la relación se presenta en la figura 5.

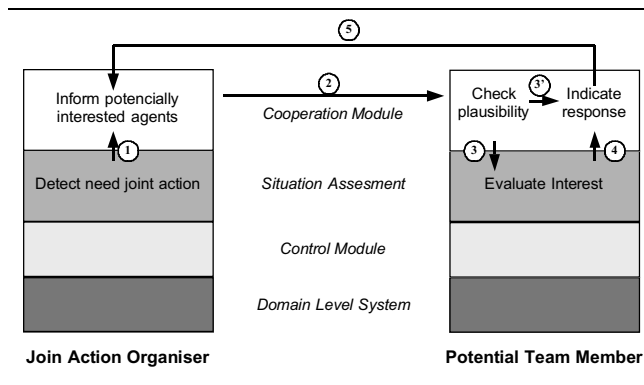


Figura 5: Establecimiento de la acción conjunta

De acuerdo a la figura, como paso inicial, el módulo de valoración de la situación informa a su módulo de cooperación que la acción social se requiere (acción 1). El módulo de cooperación identifica todos aquellos agentes del grupo de trabajo que desean colaborar con los sub-componentes del procedimiento, basado en el conocimiento de sus capacidades, que son mantenidos en el módulo de conocimiento.

Luego que todos los potenciales agentes que pueden contribuir han sido identificados, se les envía un mensaje requiriendo su participación en la acción conjunta (acción 2). Este mensaje establece que el emisor desea establecer una acción conjunta y da la razón de porque esta relación es necesaria. Cada uno de los participantes potenciales que reciben la propuesta hacen un chequeo para determinar si entiende o no la requisición.

Si la requisición es desaprobada, entonces se prepara una respuesta negativa para el organizador (acción 3'). Si la proposición es aceptada en el módulo de cooperación, ésta se pasa al módulo de valoración de la situación (acción 3) para determinar si el agente tiene suficientes recursos para acometer el problema; esto involucra el análisis de los compromisos existentes para asegurar que la nueva propuesta es consistente y puede ser desarrollada satisfactoriamente.

El resultado de la anterior evaluación puede ser "sí" o "no" y se pasa nuevamente al módulo de cooperación (acción 4) el cual devuelve la respuesta al organizador (acción 5).

Para el caso planteado, si la respuesta es afirmativa, el BAI y CSI deben decir que: a)ellos están interesados en participar el diagnóstico cooperativo b) se necesita un procedimiento común y c) Ellos usarán el modelo de cooperación de responsabilidad durante este proceso. Estos criterios se comunican explícitamente a todos los miembros de la comunidad. Hasta aquí llega la primera fase la cual estableció como objetivo la definición de los participantes y las bases para la resolución conjunta del problema.

La segunda fase del protocolo inicia con la especificación de los detalles exactos del procedimiento común. El organizador actualiza el estado de la acción conjunta a "*developing-solution*" y modifica la parte relacionada con los participantes para reflejar el hecho de que ahora se encuentran en la fase de solución. De los agentes que desean participar, el organizador selecciona algunos para minimizar el numero de miembros que el determine

como necesarios. Mediante esta estrategia se reduce el numero de procedimientos de coordinación a ejecutar. El líder del equipo entonces actualiza la parte de la contribución propuesta para indicar aquellos agentes que fueron seleccionados y aquellos que no.

Proposed Contribution:

```
((SELF ((HYPOTHESIS-GENERATION SELECTED)
        (DETAILED-DIAGNOSIS SELECTED)))
 (BAI (IDENTIFY-BOA SELECTED))
 (CSI (MONITOR-DISTURBANCE SELECTED)))
```

Figura 6: Codificación del sector de la contribución

El líder del equipo hace una propuesta inicial para los tiempos que emplean en las acciones, expresado en la parte del “Bindings” y llena los espacios de duración y de inicio y finalización de actividades.

Bindings:

```
((BAI IDENTIFY-BOA 19)
 (SELF HYPOTHESIS-GENERATION 19)
 (CSI MONITOR-DISTURBANCE 19)
 (SELF DETAILED-DIAGNOSIS 36))
```

Figura 7: Codificación de los bindings

El organizador toma cada acción del procedimiento en un orden secuencial temporalmente y acuerda con el agente concerniente el tiempo en el cual éste debe ser desarrollado. La propuesta es enviada a cada uno de los que contribuyen indicando la acción, el tiempo de inicio y la propuesta del líder para este contexto. Cada uno de los miembros evalúa la posibilidad de aceptar y si no encuentra conflictos configura la intención individual en su modelo propio tal como se muestra a continuación

Name: (ACHIEVE (IDENTIFY-BOA))
Motivation: (SATISFY-JOINT-ACTION (DIAGNOSE-FAULT))
Recipe: (IDENTIFY-BOA)
Start Time: 19 **Maximum End Time:** 34
Duration: 15 **Priority:** 5
Status: PENDING **Outcome:** (BLACK-OUT-AREA)

Figura 8: Intención individual de BAI para la identificación del área de racionamiento

Luego de establecer los acuerdos, la acción se desarrolla en el tiempo determinado y el organizador informa de la solución final. La acción conjunta es ahora operacional. El organizador cambia el estado a “*executing-joint-action*” y actualiza el listado de los participantes.

4.3 Modelo de responsabilidad usado para derivar reglas genéricas de cooperación

Cuando la acción conjunta se ha establecido, el agente debe monitorizar su ejecución. En esta fase el modelo de responsabilidad especifica las condiciones bajo las cuales el agente reconsidera sus compromisos, y describe como se debe comportar tanto localmente como con los otros agentes en caso de que haya problemas.

El modelo de responsabilidad distingue claramente cuando se deben reconsiderar los compromisos y cuando se deben desarrollar acciones, tal como se presenta en la figura 9.

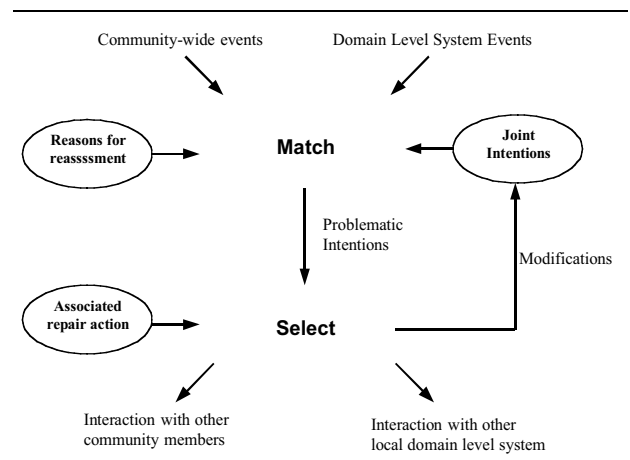


Figura 9: Esquema de responsabilidad para la ejecución de acciones conjuntas

El modulo de valoración de la situación del agente minitoriza continuamente los eventos que ocurren dentro del sistema, los cuales pueden venir desde el sistema a nivel de dominio (si la meta ha finalizado o si se necesita más información) o pueden venir desde otros agentes de la misma comunidad (Requisiciones de información, retorno de la información requerida). La mayoría de estos eventos no tienen ningún efecto sobre la acción conjunta, pero algunas circunstancias si pueden ocasionar que el agente reconsidere sus compromisos.

Entre las acciones que pueden ocasionar que el agente reconsidere su situación son: a) aquellas que le hacen perder el compromiso por la meta conjunta

y b) las que le hacen perder el compromiso por el procedimiento común

La tarea de reconocimiento de las circunstancias que llevan a la pérdida de los compromisos debe ser identificada, pero la formalización de esos eventos no está dentro del alcance del Modelo de Responsabilidad conjunta y se debe analizar el dominio para determinar claramente este tipo de eventos.

En la figura 10 aparece red de responsabilidad causal para identificación de problemas del ejemplo planteado.

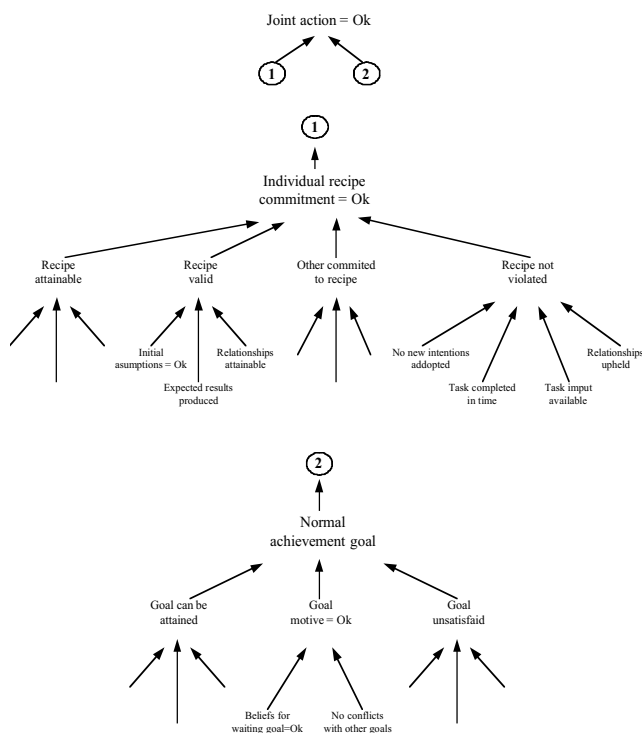


Figura 10: Red causal de responsabilidad para la identificación de intenciones problemáticas en aplicaciones de transporte de electricidad

Luego de identificar las situaciones de problema de Responsabilidad, esta se debe codificar en el modulo de valoración de la situación de GRATE*.

Los eventos forman los antecedentes de las reglas, mientras que las conclusiones de las reglas pueden ser unas de las condiciones de responsabilidad para que una nueva valoración de la acción conjunta ocurra. En la figura 11 se presentan modelos de estas reglas, las cuales enfatizan diferentes hechos tales como: Que la

meta conjunta sea satisfecha, que la motivación para la meta conjunta este presente, que el procedimiento común haya sido violado y que el procedimiento común sea invalido.

R1 match: if task t has finished executing and
t has produced the desired outcome of the
joint action
then the joint goal is satisfied

R2 match: if receive information i and
i is related to the triggering conditions
for joint goal G and
i invalidates beliefs for wanting G
then the motivation for G is no longer present

R3 match: if delay task t1 and
t1 is a component of the common recipe R for
a joint action and
t1 must be synchronised with task t2 in R
then R is violated

R4 match: if finished executing common recipe R and
expected results of R not produced and
alternative recipe exists
then R is invalid

Figura 11: Reglas de condiciones de responsabilidad

Si una regla evidencia algún tipo de peligro para la acción conjunta, entonces se identifica este problema y se busca una solución la cual varia entre abandonar las acciones hasta reorganizar las tareas. Esta circunstancia se representa con las reglas que se presentan a continuación.

R1 select: if joint goal is satisfied
then abandon all associated local activities
inform cooperation module that the joint
action has successfully finished.

R2 select: if motivation for joint goal is no longer
present
then abandon all associated local activities
inform cooperation module that the
motivation for the joint
action is no longer present

R3 select: if common recipe R is violated and
R can be successfully rescheduled
then suspend local activities associated
with R
reset the descriptions and timings of R
inform cooperation module that R has
been violated and propose new
timings

R4 select: if common recipe R1 is invalid and
alternative recipe R2 exists
then abandon local activities associated
with R1
inform cooperation module that R1 is
invalid and propose R2 as an
alternative course of action

Figura 12: Reglas de solución de problemas

Luego que se han tomado algunas acciones locales tales como el abandono, suspensión o reorganización de actividades, el Modelo de

Responsabilidad establece que se comunique todo lo sucedido a los otros agentes. Esto último se realiza utilizando el módulo de cooperación basado en la información dada por el modelo de valoración de la situación. Las respuestas dadas por el modulo de cooperación se presentan en la figura 13.

```
R1 inform:if joint action has successfully finished
           then inform all team members of successful completion
           see if results should be disseminated outside the
           team

R2 inform:if motivation for joint goal G is no longer present
           then inform other members of the team that G needs to
           be abandoned

R3 inform:if common recipe has been violated and
           new timings proposal exists
           then inform other team members of violation and also
           propose new timings as a means of fixing the
           problem

R4 inform:if common recipe R1 is invalid and
           alternative proposal R2 exists
           then inform other team members that R1 is invalid and
           offer R2 as an alternative
```

Figura 13: Reglas para comunicación de agentes

5. CONCLUSION

La mayoría de sistemas distribuidos de inteligencia artificial no tienen una representación explícita de la actividad de cooperación. Mediante la adopción de la aproximación basada en la responsabilidad, muchas de las asunciones tácitas acerca del proceso de resolución cooperativa del problema que puede soportar las reglas al nivel de superficie, están explícitamente disponibles para los agentes GRATE*. Lo anterior significa que los agentes basan su razonamiento en un modelo con un principio de cooperación y por eso puede responder de manera más flexible y robusta al comportamiento anormal de aplicaciones complejas.

6. BIBLIOGRAFIA

N. R. Jennings. Controlling Cooperative Problem Solving in Industrial Multi-Agent Systems using Joint Intentions. Department of Electronic Engineering, Queen Mary and Westfield College, University of London. University of London, Mile End Road, London E1 4NS, UK. 1995