

DESARROLLO DE UN SISTEMA DE
SUPERVISIÓN DE SISTEMAS DE
PRODUCCIÓN CONTINUA.

por

Amoa Torrealba Rojas

Trabajo de grado presentado como requisito parcial para obtener el grado de Magíster

Scientiae en Automatización e Instrumentación

Universidad de Los Andes

Mérida, Venezuela

2005

Tutor: Dr. Edgar Chacón R.

TABLA DE CONTENIDO

Tabla de contenido.....	i
Indice de tablas y figuras	iii
CAPÍTULO I	1
Introducción	1
Aspectos Generales	1
Justificación	4
Antecedentes	6
Planteamiento del problema.....	7
Hipótesis	8
Objetivos.....	9
Delimitaciones.....	9
Organización del trabajo	10
CAPÍTULO II	11
Bases teóricas para los sistemas de producción holónicos	11
El proceso de producción.....	11
Sistemas de Manufactura Holónica	13
HoMuCS.....	17
Coordinación / Optimización de Producciones Complejas	19
Sistemas Holónicos vs Sistemas Multi Agente	24
CAPÍTULO III	27
Sistemas multiagentes.....	27
Agentes	27
Características	28
Estándares para los agentes.....	32
Organismos de estandarización	33
KSE.....	33
The Agents Society	33
OMG.....	34
FIPA.....	34
FIPA.....	34
Descripción General de FIPA	34
Declaración de intenciones de FIPA.	35
Organización del estándar	35
Referencias Tecnológicas.	36
Referencias a Aplicaciones (Informative Specification)	36
El lenguaje de comunicación de los agentes.	38
Introducción a los lenguajes de comunicación de agentes.....	38
El lenguaje ACL.....	39
Definición del tipo de mensaje que se envía	40

Participantes de la comunicación.....	41
Contenido del mensaje	42
Descripción del contenido	43
Control de conversación.	44
Protocolos de interacción FIPA	45
FIPA Request Interaction Protocol.....	46
FIPA Query Interaction Protocol.....	47
FIPA Request When Interaction Protocol	48
FIPA Contract Net Interaction Protocol	49
FIPA Iterated Contract Net Interaction Protocol.....	51
FIPA English Auction Interaction Protocol	53
FIPA Dutch Auction Interaction Protocol.....	54
FIPA Brokering Interaction Protocol	56
FIPA Recruiting Interaction Protocol.....	58
FIPA Propose Interaction Protocol.....	60
Herramientas para la implementación de agentes	61
JADE	62
Agentes del sistema.....	66
 CAPÍTULO IV	 69
Metodología y Diseño	69
Vista general sobre la extracción del petróleo crudo de un pozo	70
Unidad de Producción.....	76
Capa Física de la UP	77
Modelado del Pozo	77
Control de flujo de Salida del Pozo.....	79
Capa de Decisión e Información de la UP.....	80
Estados discretos para las bombas.....	82
Estados discretos del tanque reservorio.....	84
Estados discretos de la tubería.....	86
Estados discretos de la válvula	87
Estados discretos de la Energía.....	88
Control discreto de las bombas.....	89
Estados discretos de la producción del proceso	92
Comportamiento de una Unidad de Producción	94
Agentes	96
Agente de Control	98
Agente de Recursos.....	98
Agente Supervisor de UP.....	99
Agente Negociador	99
Agente Supervisor de Zona	100
Comunicación inter agentes en la UP.....	101
 CAPÍTULO V	 109
Análisis de los Resultados	109

CAPÍTULO VI	113
Conclusiones y Recomendaciones	113
BIBLIOGRAFÍA	115

INDICE DE TABLAS Y FIGURAS

Figura 1: Inteligencia y precisión en un sistema jerárquico.....	12
Figura 2: Intercambio de conocimientos dentro de los sistemas de manufactura holónica. 16	
Figura 3: Las clases básicas de la arquitectura HoMuCS.....	18
Figura 4: Modelo UML para una unidad de producción.....	21
Figura 5: Componentes del sistema de control Jerárquico.....	23
Figura 6: Comparación entre autonomía y cooperación para diferentes entidades	25
Tabla 1: Parámetros del mensaje.....	39
Tabla 2: Tipos de actos de comunicación.....	40
Figura 7. Protocolo FIPA – Request	47
Figura 8: Protocolo FIPA – Query.....	48
Figura 9. Protocolo FIPA Request When	49
Figura 10. Protocolo FIPA Contract Net	51
Figura 11. Protocolo FIPA Iterated Contract Net.....	52
Figura 12. Protocolo FIPA English Auction.....	54
Figura 13. Protocolo FIPA Dutch Auction	56
Figura 14: Protocolo FIPA Brokering.....	58
Figura 15: Protocolo FIPA Recruiting	60
Figura 16: Protocolo FIPA Propose	61
Figura 17. Ventana en JADE de un mensaje ACL.....	63
Figura 18. Ventana en JADE del agente manejador remoto (RMA).....	65
Figura 19. Ventana en JADE del agente de páginas amarillas (DF).....	66
Figura 20. Ventana en JADE del Agente Dumy.....	67
Figura 22: Descripción de la Unidad de Producción.....	70
Tabla 3: Producción promedia por pozo en diferentes países	75
Figura 23: Proceso simplificado de extracción de crudo de un pozo de petróleo.	78
Figura 24: Control a través del PID en lazo cerrado.	79
Figura 25: Control del Flujo de Salida del Pozo.....	80
Figura 26: Arquitectura de Supervisión.....	81
Figura 27: Red de Petri para los estados de la bomba	83
Tabla 4: Descripción de la Red de Petri para los estados discretos de la bomba.	83
Figura 28: Red de Petri para los estados del tanque reservorio.....	85
Tabla 5: Descripción de la Red de Petri para los estados discretos del tanque reservorio ...	85
Figura 29: Red de Petri para los estados de la tubería.....	86
Tabla 6: Descripción de la Red de Petri para los estados discretos de la tubería	87
Figura 30: Red de Petri para los estados de la válvula.	87
Tabla 7: Descripción de la Red de Petri para los estados discretos de la válvula.....	88

Figura 31: Red de Petri para los estados de la Energía.....	88
Tabla 8: Descripción de la Red de Petri para los estados discretos de la Energía.....	89
Figura 32: Capa donde se lleva a cabo el control discreto de las bombas	90
Figura 33: Diagrama de flujo para el control discreto de las bombas.....	91
Figura 34: Red de petri para los estados de la producción	92
Tabla 9: Descripción de la Red de Petri para los estados discretos de la producción	93
Figura 35: Comportamiento de una unidad de producción genérica [3].	95
Tabla 10: Descripción de la red de petri para los nodos del comportamiento de una UP....	96
Figura 36: Agentes que componen la UP y sus comunicaciones.....	97
Figura 37: Interacción entre el agente supervisor de zona y las UPs pertenecientes a la zona.	100
Figura 39: Comunicación inter agentes	107
Figura 40: Interfaz Gráfica para el Agente Supervisor Global	109
Figura 41: Interfaz Gráfica para el Agente Supervisor de la Unidad de Producción	110
Figura 42: Interfaz Gráfica para el Agente de Control	110
Figura 43: Interfaz Gráfica para el Agente de Recursos	111
Figura 44: Interfaz Gráfica para el Agente Negociador	111

CAPÍTULO I

INTRODUCCIÓN

Aspectos Generales

El extenso desarrollo y la expansión de la tecnología de información se han convertido en una parte integral en la vida cotidiana de la sociedad moderna, por lo cual podríamos afirmar que nos encontramos en la era de la informática.

Hemos experimentado la incursión de la tecnología de información en todas las áreas, desde las aplicaciones cotidianas y simples a los sistemas de información complejos. Esto produce un cambio profundo en nuestro ambiente y estilo de vida ya que las personas, al igual que los mercados de negocios y las industrias, están volviéndose más dependientes de esa tecnología de información.

Por otro lado, existe una tendencia a la globalización en los mercados. Los clientes y los proveedores ya no están restringidos a los límites geográficos o físicos, porque tienen acceso, prácticamente instantáneo, a la información comercial del mundo, debido en gran parte, a la Internet.

Al querer entrar a competir en los mercados, no solo conviene hacer que el producto sea de buena calidad, sino que también tenga un precio accesible al consumidor y para ello, se debe optimizar la producción sobre la base de los costos, calidad y beneficios, entre otros.

Muchas veces la disminución de los costos de producción, se lleva a cabo gracias a la fabricación en masa de los productos que los clientes requieren. Para realizar este tipo de fabricación, se necesita determinar los elementos necesarios para hacerlo y uno de estos es la automatización.

En forma general, la automatización, incluye algún sistema informático o de computación, capaz de llevar a cabo ciertas tareas asignadas. En algunos casos, la automatización ayuda a reducir los costos de producción dejando el volumen de producción constante, en otros simplemente permite aumentar, de forma flexible, el volumen de

producción al mismo costo, lo cual se traduce, entre otras cosas, en la capacidad de competir en el mercado, obteniendo beneficios más altos.

La producción en masa involucra diferentes tipos de procesos, dos de ellos son: el proceso por lotes o intermitente (manufactura) y el proceso continuo. En el primero los materiales no circulan en forma constante, sino en pasos discretos y secuenciales que agregan valor al producto en la cadena de producción. En el segundo, los materiales y/o las diversas partes deben circular a lo largo de la producción en cantidad constante en un período de 24 hrs. los 365 días al año y la producción no debe paralizarse ya que las paradas de la línea por cualquier razón suelen ser caras.

En los procesos de producción continuos existen grandes cantidades de variables y/o datos que deben ser analizados constantemente, con el fin de prevenir fallas, estancamientos o paradas en la línea de producción, para ello se utilizan sistemas informáticos o de computación que se encargan de estudiar los datos de las variables y controlarlas en forma automática.

Esta automatización, no solo ayuda a prevenir y corregir fallas en la producción de procesos continuos, sino que también podría ayudar en la optimización del proceso, si lograra comportarse como un sistema “inteligente”, con lo cual se ahorraría el costo de tener que contratar recursos humanos especializados para llevar a cabo esta tarea.

En este trabajo se pretende crear un sistema automatizado, basado en la teoría holónica, que optimice la producción de un proceso continuo. Este sistema debe ser capaz de supervisar el proceso y de decidir cuales serán los criterios a seguir para optimizar la producción.

Acá se ha tomado como ejemplo de una producción continua: la producción de petróleo, la cual involucra una serie de actividades complejas como son: la extracción, transportación y el almacenamiento del mismo.

En general, la producción de petróleo es un sistema complejo y difícil de automatizar, debido a la gran cantidad de unidades que lo componen, el comportamiento de cada una de ellas, las interrelaciones existentes y la gran cantidad de actividades involucradas. Para intentar resolver este problema se hace una aproximación del sistema, la cual, consiste en

simplificar y separar las funciones de producción, según su fundamento, en elementos o unidades de producción con funciones que le sean propias (holones). Estos elementos, deberán trabajar de una manera integrada y coordinada para garantizar que el beneficio de la producción sea óptimo. La aproximación ayuda a reducir la complejidad del sistema de producción, para poder abordarlo de forma cómoda. Así que cada elemento del sistema, en la aproximación, se describe como un holón y el conjunto completo como una holarquía o sistema holónico. Dentro de esta holarquía, cada holón interactúa con los otros para contribuir a la optimización y cumplimiento de los objetivos del sistema holónico.

Un holón es un elemento dentro del sistema, que tiene propósitos específicos y por lo tanto debe tener cierto grado de autonomía que le permita tomar decisiones para lograr sus propósitos internos. También es capaz de comunicarse con los otros elementos del sistema para informarles de su propio estado y/o interactuar con ellos para ayudar en el cumplimiento de los objetivos propios o de algún miembro con el que interactúe. Cuando un holón cumple con sus propios objetivos también está contribuyendo al cumplimiento del objetivo global de la holarquía.

El holón posee noción sobre sí mismo, ya que conoce sus necesidades y limitaciones, en consecuencia si tuviera alguna necesidad, deberá ser capaz de entablar un proceso de negociación con los otros elementos para solucionarlo. Ahora bien, “... *la arquitectura holónica depende de la tecnología de agentes...*” [5] para comunicarse, por lo tanto cada holón tendrá asociado, a uno o más agentes de software que se encargarán de ayudar en el intercambio de información entre Holones, es decir, todos los procesos de negociación serán realizados por los agentes de software en representación de cada holón.

La meta principal de este trabajo, es desarrollar un supervisor de un sistema de producción continua, visto desde el punto de vista holónico para intentar disminuir la complejidad del mismo, a través de un sistema de eventos discretos, que representará los diferentes estados del sistema continuo. Éste supervisor debe garantizar que la meta de la holarquía se cumplirá para conseguir el máximo beneficio de la producción.

Toda la holarquía debe actuar sobre la base de una agenda de producción común, que es simplemente un registro de los estados discretos administrado y distribuido por el

supervisor, ésta puede ser construida usando la información clave con respecto a las condiciones actuales bajo las que trabajan las diferentes unidades de producción. Esta información consiste en parámetros locales como: el estado y la capacidad de las unidades de producción, condiciones financieras, suministros y productos, etc., y debe ser actualizada en todo momento en la agenda de producción, para que a través de ésta, se determinen las actividades a realizar. Estas actividades deben ser realizadas de acuerdo con la meta global y el estado real de las unidades de producción, y deben ser implementadas localmente sobre las infraestructuras de computación existentes. Al desarrollar un sistema basado en la agenda de producción que supervise el proceso, se garantiza el cumplimiento de la meta del sistema holónico.

Justificación

Casi siempre los sistemas industriales se han visto como sistemas estáticos, donde las cosas, si cambian, lo hacen de forma paulatina y predecible, pero ¿qué pasa cuando algo inesperado sucede? Normalmente lo que sigue, es el caos.

Este tipo de ambiente puede ser considerado como un ambiente dinámico, un ambiente donde las cosas están cambiando constantemente, agregando a éste, el hecho de que este tipo de ambiente es típicamente discreto, es decir, los hechos ocurren de forma puntual y esta ocurrencia es de naturaleza aleatoria. Ante esto, se vuelve evidente que es, de hecho, un ambiente complicado.

Se puede asegurar que siempre habrá cambios, como por ejemplo: la tendencia del mercado, gustos de los consumidores, cambios en la demanda de los productos, avería de alguna máquina, introducción de un nuevo producto o de una nueva tecnología, etc., por lo tanto, se debe de tomar las medidas para estar preparados ante éstos. Una solución, consiste en hacer que estos sistemas, sean ágiles y se adapten a los cambios de forma rápida y eficiente.

Esto permitiría poder hacer ajustes de los elementos individuales en forma independiente, sin tener que perturbar el sistema general, y además, permitiría dinamizar los cambios en el sistema con menos esfuerzo y menos configuraciones, lo cual aseguraría la

sobre vivencia del mismo, siempre y cuando las adaptaciones sean hechas en forma ágil, coherente y rápida.

Al tomar en cuenta lo anterior, surge la necesidad de reimponer la manera en que se desarrollan los sistemas industriales y se operan, ya que muchas técnicas de hoy en día están expuestas a ser inválidas, no-óptimas o incluso imprácticas.

El desarrollo de éste trabajo representa un gran reto, ya que se pretende enfocar la automatización de un sistema industrial continuo desde el punto de vista holónico, lo cual representa un nuevo paradigma en este tipo de sistemas.

Con el enfoque holónico, se pretende, que este sistema industrial, en particular, sea ágil; así que si surgiera algo inadvertido, se podrían hacer cambios a los elementos individuales para adaptarlos a las condiciones que se estén presentando sin que se derrumbe todo el sistema, lo cual asegura la supervivencia del mismo. Además éste tipo de enfoque asegura que los elementos holónicos son formas estables que sobreviven a los disturbios, es decir, que son capaces de manejar ciertos cambios y saben lo que deben hacer, pero también, son las formas intermedias que proporcionan la funcionalidad apropiada para el todo más grande.

Este enfoque holónico de la automatización, puede implementarse dentro de la infraestructura de computación existente, lo cual representa un ahorro dentro de los gastos que pudieran presentarse en la inversión inicial. Por otra parte, esto asegura, que si en el futuro, se tuviera que hacer ajustes, los mismos se harían, sobre la infraestructura inicial, con costos mínimos.

Al tomar en cuenta lo anterior, se podría asegurar, que el sistema industrial del futuro deberá consistir en unidades elementales simples (fractales, holones, etc.) ya que son modulares, flexibles, estables, autónomos, convergentes, dinámicos e inteligentes. Estas unidades deben existir en un sistema donde ellas actúen recíprocamente al ritmo de las metas globales y, paralelamente, sigan con sus propias metas de una manera coordinada, lo cual asegurará la supervivencia del sistema

Antecedentes

La teoría Holónica fue propuesta por Koestler, apoyándose en una conclusión hecha por Herberto Simon quien afirma que los sistemas complejos se desarrollarán de sistemas simples mucho más rápidamente si hay formas intermedias estables que si no las hay, trayendo como consecuencia que los sistemas holónicos sean sistemas Jerárquicos.

Por otra parte existen teorías similares como por ejemplo: “The Holonic Manufacturing System” (HMS) teoría desde una perspectiva técnico social, “The Fractal Factory” teoría desde una perspectiva matemática y “The Biological Manufacturing System” (BMS) teoría desde una perspectiva biológica. Todas estas teorías implican conclusiones más o menos similares llamadas como un sistema de manufactura HoMuCS la cual consiste en unidades elementales (fractales, holones, células biológicas) que son autónomas, cooperativas, dinámicas y recientemente inteligentes. Estas unidades deben existir en un sistema donde ellas interactúan en orden para cumplir con las metas globales mientras siguen sus propias metas de una manera coordinada. Ello permite el cambio constante, y además los cambios pueden hacerse independientemente a los elementos individuales sin perturbar el sistema. Esto también permite cambios dinámicos al sistema con menos esfuerzo y menos configuraciones.

No obstante las teorías anteriores [1, 5, 7, 8, 9] siempre han estado orientadas hacia los procesos de manufactura, recientemente se han hecho algunas investigaciones en sistemas de producción continua [2, 3, 4] que proponen esquemas de control para una automatización integrada, bajo perspectivas muy similares a las de manufactura, tales como: unidades elementales autónomas, cooperativas, etc. Esta automatización va más allá del control clásico ya que también engloba aspectos de planificación, gerencia, optimización de recursos materiales y humanos, etc.

El presente trabajo, plantea la continuación y desarrollo del artículo “**Coordination / Optimization of production complexes**” [3], en el cual se utiliza la teoría de sistema a eventos discretos, SED, para poder transformar estados continuos de la planta en eventos discretos, de forma tal que se pueda describir toda la dinámica interior de la unidad de producción de petróleo, a través de la teoría holónica.

Al enfocar el proceso de producción de petróleo de esta forma se reduce la complejidad del sistema, es decir, el proceso se separa en pequeños componentes llamados unidades de producción los cuales, a su vez, representan partes de algún proceso físico/químico.

Por otra parte, al separar las unidades de producción en representación de algún sistema físico/químico se garantiza la estabilidad de las formas intermedias que componen el sistema total de producción, ya que *“Los sistemas complejos se forman más fácil, si están compuestos de formas / subsistemas intermedias estables, que si no lo están...”* *“En estos aspectos estructurales no es una agregación de partes elementales, y en su aspecto funcional no es una cadena de unidades de comportamiento elementales.”* [5].

De la misma manera, se debe asegurar que cada una de estas unidades de producción sea autónoma y pueda interactuar con las otras unidades para lograr la meta en común; de allí nace el concepto de Unidad de Producción Autónoma: *“... que es un elemento que es capaz de auto manejarse y de informar a las otras unidades de su propio estado.”* [3]. Ellas son consideradas autónomas porque están capacitadas para tomar sus propias decisiones, por medio de su dispositivo de control local, para mantener el sistema seguro y trabajando de acuerdo a las especificaciones recibidas del supervisor. Igualmente, *“... las especificaciones recibidas del supervisor resultan de la evaluación del estado de los subsistemas, en términos de capacidad, actuación, disponibilidad, relaciones entre ellos, estrategias de producción y factores externos, etc.”* [3].

Lo desarrollado en estos trabajos establece el basamento teórico que sirve como principio en el desarrollo de esta investigación, ya que constituye el antecedente de los planteamientos utilizados y expuestos acá.

Planteamiento del problema

Anteriormente hemos dicho que la teoría holónica pretende hacer que los sistemas industriales, sean ágiles, de forma tal que si surgiera algo inadvertido, se podrían hacer cambios a los elementos individuales, para adaptarlos a las condiciones que se estén presentando, sin que se derrumbe todo el sistema.

Por otra parte también se ha dicho que para reducir la complejidad de los sistemas se puede hacer una aproximación, que simplifique y separe las funciones de producción, según algún proceso físico/químico, en elementos o unidades de producción que lleven a cabo las metas que les hayan sido especificadas. Esta simplificación utilizaría la teoría de sistema a eventos discretos, SED, y la teoría de holones para describir toda la dinámica interior de la unidad de producción de petróleo, ayudando a reducir la complejidad.

Las unidades de producción también deben de tener dos características principales: *que sean autónomas*, es decir que tengan ciertos grados de independencia para que puedan manejar algunos contratiempos sin pedir autorización a unidades superiores y *que sean cooperantes* es decir, que interactúen en orden para cumplir con las metas globales mientras siguen sus propias metas de una manera coordinada.

Sin embargo, al ser este enfoque un nuevo paradigma en los sistemas continuos, surgen muchas dudas como por ejemplo:

¿Es realmente el enfoque holónico adecuado para un sistema de producción continuo?

¿Es factible implementar éste enfoque en los sistemas continuos, desde el punto de vista técnico?

¿Se puede garantizar que el sistema va a ser capaz de adaptarse a los cambios imprevistos?

¿Sería capaz el sistema de asegurar una producción óptima?

¿Cómo implementar la autonomía y la cooperación holónica para que cumplan con las metas globales mientras sigue con sus propias metas?

Hipótesis

Un sistema de producción puede ser supervisado si su estado es discreto y los objetivos (agenda) de producción se obtienen de una negociación que tome en cuenta la autonomía de las unidades de producción bajo un enfoque holónico.

Objetivos

- Evaluar los sistemas multiagentes como herramientas para la implementación de la teoría de holones en los sistemas continuos.
- Definir una estrategia de supervisión apropiada para garantizar la optimización.
- Definir cuales son los estados y las variables convenientes, para evaluar las unidades de producción, tales que, le permitan al supervisor garantizar la coordinación de los elementos y el logro de la meta global.
- Definir los protocolos adecuados de comunicación entre elementos del sistema holónico, que se ajusten a las necesidades de un sistema de producción continuo.

Delimitaciones

Este trabajo está centrado, en la evaluación de los sistemas multiagentes como instrumentos computacionales que permiten comprobar la hipótesis y la aplicación de la teoría holónica. Dado que los sistemas multiagentes se caracterizan por su interacción en una red de computadoras, pero de forma transparente para el desarrollador, no se estima necesario la descripción del funcionamiento de estos en todas capas de la comunicación debido a que existen aplicaciones intermedias que se encargan de esto en forma muy estable (middleware) limitando el presente trabajo al diseño e interacción del sistema holónico y el supervisor en las capas de aplicación y presentación.

La capa de presentación define cómo representar los datos de computación de diferentes tipos, así como también el protocolo que se necesita para trasladar los datos, desde y hacia, el nodo local.

Por su parte la capa de aplicación da las especificaciones para las aplicaciones usando la red: cómo enviar una petición, cómo especificar un nombre de archivo sobre la red, cómo responder a una petición, etc.

Por otra parte el sistema de producción continuo escogido para intentar comprobar la hipótesis, es el de la producción de petróleo. Por su naturaleza, este sistema define la producción como una serie de actividades de diversos tipos, como son, la exploración, la extracción, transportación, refinación, etc., del petróleo. Sin embargo, para este trabajo solo

se toma en cuenta la extracción en la producción primaria, es decir, cuando el crudo dentro de un yacimiento se encuentra bajo una alta presión. Esta extracción en la producción primaria es representada por un modelo simplificado en una simulación por computadora, ya que resulta muy difícil obtener acceso a un sistema físico real y a los datos reales manejados por nuestra principal industria petrolera.

Es dentro de estos límites que se desarrolla el trabajo, los cuales se consideran aptos para la comprobación de la hipótesis.

Organización del trabajo

El trabajo se organiza en VI capítulos. Este capítulo presenta los objetivos generales del trabajo dentro del contexto de la optimización de la producción desde el punto de vista holónico, lo cual es la razón principal de esta investigación.

El segundo capítulo establece las bases teóricas para sistemas de producción situándolos dentro de los sistemas Holónicos. Introduciendo los conceptos de sistemas holónicos de producción, para manufactura y sistemas continuos, y su relación con los sistemas multi-agentes, negociación entre agentes, y los estándares necesarios para la comunicación.

El tercer capítulo describe en forma amplia los sistemas multi agente, los organismos de estandarización de agentes, estándares necesarios para la comunicación entre ellos, protocolos de comunicación, negociación, etc. También se describe en forma general a JADE como la herramienta a utilizar para la programación de los agentes.

El cuarto capítulo está orientado a la descripción de la forma de implantación de sistemas holónicos y sistemas multi agente en un ejemplo ideal y simplificado de una producción continua, como lo es, la extracción de petróleo.

El quinto capítulo muestra los resultados obtenidos de la programación del sistema holónico implementado en el ejemplo de producción continua

Por último el sexto capítulo muestra las conclusiones en base a los objetivos planteados al inicio de este trabajo y algunas sugerencias para trabajos futuros.

CAPÍTULO II

BASES TEÓRICAS PARA LOS SISTEMAS DE PRODUCCIÓN HOLÓNICOS

En éste capítulo se intenta presentar la base teórica necesaria para situarnos dentro del contexto de los procesos de producción y cómo se relacionan con los sistemas de producción holónica (continua y de manufactura). Así mismo se intenta dar la relación existente entre los sistemas de producción holónica con los sistemas multi agentes, los cuales son quienes, en definitiva, ofrecen las herramientas adecuadas, presentadas también en este capítulo, para llevar a cabo la implementación de sistemas de producción bajo la visión holónica.

El proceso de producción

Cuando se habla del proceso de producción, debe incluirse no solo la transformación de la materia prima, la secuencia de la producción y el diseño de los productos, ligados directamente a las actividades de producción, sino también a las actividades de mercadeo, finanzas, personal e ingeniería.

Para tomar en cuenta lo anterior, se debe contar con las herramientas adecuadas que faciliten todas las etapas del proceso de producción, como por ejemplo un sistema de producción automatizado, es decir, *“...aquel donde la producción es principalmente coordinada y controlada mediante sistemas automáticos.”* [4].

Los procesos de manufactura difieren de los de producción continua en que los primeros se describen mediante un sistema de transiciones mientras que los segundos son descritos a través de ecuaciones diferenciales. Otra diferencia se basa en los modos de producción, los métodos de control y los dispositivos requeridos. Por lo tanto, la automatización de cada uno de ellos, exige métodos y herramientas diferentes adaptados a cada caso de acuerdo a sus características generales. Sin embargo, en ambos casos, el proceso de automatización es jerárquico, con funciones específicas en cada nivel de la jerarquía.

La cantidad de información necesaria y el nivel de precisión dependen del nivel de la jerarquía en la automatización. La precisión en los niveles inferiores es mucho mayor, tanto

en tiempo, como a nivel de detalle, sin embargo el contenido de la información es más simple.

La siguiente figura tomada de [4] nos muestra como a medida que se está mas cerca de los niveles inferiores, la inteligencia del sistema de producción disminuye y la precisión aumenta, es decir, en la planificación de un objetivo se presenta una mayor complejidad en el procesamiento de los datos y aumenta el volumen de datos necesarios para la realización de la misma, por lo tanto se requiere de mayor inteligencia.

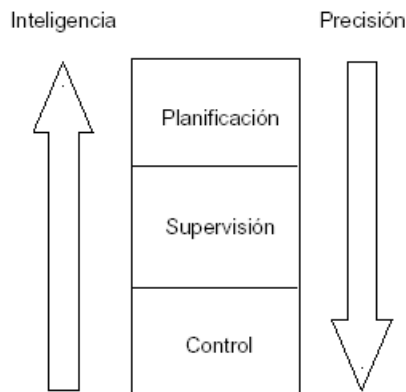


Figura 1: Inteligencia y precisión en un sistema jerárquico.

Dentro de los procesos de producción tenemos los siguientes:

Proceso de Manufactura: En éstos procesos, las principales funciones son de maquinación de piezas y ensamblaje; la maquinación forma productos intermedios que luego serán ensamblados. Este tipo de proceso lo encontramos, por ejemplo, en la fabricación de carros, maquinarias y partes automotrices.

Procesamiento por lotes: acá, un proceso recibe un conjunto de materia prima, la cual es transformada en un nuevo producto mediante mezclas y aplicación de energía. Una vez terminado el proceso, éste arranca nuevamente para repetir el mismo o realizar uno nuevo. Este tipo de proceso se encuentra en la industria siderúrgica, en la industria de alimentos y en gran parte de la industria química.

Procesos continuos: son los procesos de transformación o producción que trabajan de manera permanente. Como ejemplo de los mismos están: las plantas generadoras de energía eléctrica, la producción y refinación de petróleo y el manejo del gas. Algunas plantas procesan productos que son utilizados aguas abajo por otra planta. La mayoría de estos sistemas son considerados sistemas complejos, por el tipo de transformaciones que se dan y por la interrelación entre los procesos que están presentes en el sistema.

Uno de los principios utilizados en la automatización de sistemas de producción es el de mantener los procesos controlados de manera autónoma. Esto conlleva a que sistemas con múltiples unidades, deben de tener cada una su sistema de control. También es necesario mantener una información común entre ellos, que asegure un óptimo desempeño global, para lo cual tienen que estar interconectados y de esa manera transmitir información entre ellos acerca de su estado de operación, para lograr lo anterior existe una nueva propuesta llamada Sistemas de Producción Holónicos, tanto para la producción de manufactura como para la producción continua.

Sistemas de Manufactura Holónica.

El sistema de manufactura holónica conocido también por sus siglas en inglés HMS se enfoca específicamente, en el campo de control de procesos de manufactura, en las tecnologías de información en manufactura, y en satisfacer la necesidad de fabricar poco volumen y gran variedad de productos. También provee un proceso de manufactura dinámico y descentralizado, en el cual los humanos son integrados consiguientemente.

Su concepto principal es el término Holón, cuyo origen proviene del trabajo de Koestler. “Un Holón se define simultáneamente como un todo y como una parte del todo. Así, un Holón puede componerse de otros holones. Es autónomo y cooperativo y a veces inteligente. Tiene características híbridas de entidades que cooperan con otras entidades ofreciéndoles sus servicios (servidores) y al mismo tiempo pueden actuar autónomamente como entidades libres (agentes libres). Cada elemento del sistema de la producción puede ser

un Holón; además, estos holones cooperan entre sí. A un sistema de Holones se le llama Sistema Holónico.” [6].

El punto fuerte de la organización holónica, u holarquía, es que permite la construcción de sistemas complejos que son eficientes en el uso de los recursos, altamente resistentes a disturbios (internos y externos), y adaptables a los cambios en el ambiente en el cual existen. El paradigma de HMS busca trasladar este concepto, que Koestler desarrolló para las organizaciones sociales y organismos vivientes, en un conjunto de conceptos apropiados para la manufactura industrial. El concepto combina los mejores rasgos de las organizaciones jerárquicas y heterárquicas, ya que preserva la estabilidad de una jerarquía mientras provee la dinámica flexible de la heterarquía.

A continuación, se muestra una lista de definiciones principales que ayudan a entender y a orientar la versión de los conceptos holónicos dentro del ambiente de manufactura.[9]

Holón una parte autónoma y cooperativa del sistema de manufactura que sirve para transformar, transportar guardar y/o validar información y objetos físicos. El holón consiste en una parte del proceso de información y otra parte del proceso físico. Un holón puede ser parte de otro holón.

Autonomía La capacidad de una entidad para crear y controlar la ejecución de sus propios planes y/o estrategias.

Cooperación Un proceso con el que un conjunto de identidades desarrollan planes mutuamente aceptables y ejecutan esos planes.

Holarquía Un sistema de holones que pueden cooperar para cumplir con una meta u objetivo. La holarquía define las reglas básicas para la cooperación de los holones y, por consiguiente limita su autonomía.

Sistema de Manufactura Holónica (HMS) es una holarquía que integra a todos los rangos en las actividades de manufactura, desde registrar una orden, hasta el diseño, producción y mercadeo, que convienen a las empresas de manufactura ágiles.

Atributos holónicos el atributo de una entidad que la hace un holón. Los atributos mínimos son la autonomía y la cooperación.

Un sistema holónico de manufactura u holarquía consiste en tres tipos básicos de holones, llamados holón recurso, holón producción y holón orden de producción.

Holón Recurso (RH) contiene partes físicas, llamadas recurso de producción de los sistemas de manufactura, y una parte del proceso de producción que controla el recurso. Este holón, ofrece la capacidad y funcionalidad a los holones circundantes. Tiene la habilidad para localizar los recursos de producción, y el conocimiento y procedimientos para organizar, usa y controla los recursos de producción para manejar la producción. Un Holón Recurso es una abstracción de medios de producción como una fábrica, una tienda, las máquinas, los hornos, las tuberías, las paletas, las herramientas, el almacenamiento de material, el personal, la energía, etc.

Holón Producto (PH) mantiene el proceso y el conocimiento del producto para asegurar la fabricación correcta del producto con la calidad suficiente. El Holón Producto contiene información consistente y actualizada sobre el ciclo de vida del producto, los requisitos del usuario, diseño, planes del proceso, factura de materiales, garantía de los procedimientos de calidad, etc. Así como también contiene, el “modelo del producto” del tipo de producto. El Holón Producto actúa como un servidor de información para otros holones en el HMS.

Holón Orden (OH) representa una tarea en el sistema de manufactura. Es responsable de realizar el trabajo asignado, correctamente y a tiempo. Maneja el producto físico que está produciéndose, la información del estado del producto, y todo lo relacionado a la información logística del proceso. Un Holón Orden puede representar las órdenes del cliente, hacer la acción de las órdenes, hacer órdenes para mantener y reparar los recursos, etc.

Los holones RH y OH denotan el material y los recursos en la fábrica mientras que PH denota la preparación de las actividades de producción y los procesos de información derivados.

La siguiente figura, tomada de [6] muestra las relaciones existentes entre los tres holones anteriores.

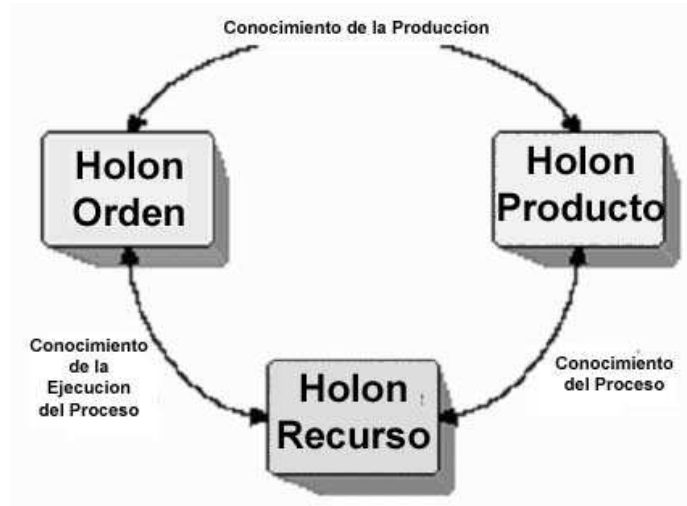


Figura 2: Intercambio de conocimientos dentro de los sistemas de manufactura holónica.

El holón producción, y el holón orden tienen el conocimiento de la producción, esto es, los métodos y la información con respecto a cómo ejecutar cierto proceso en cierto recurso. Conocen la capacidad de los recursos, cuales procesos pueden ejecutar, los parámetros de los procesos relevantes, la calidad del proceso, etc.

El holón orden intercambia conocimientos de la ejecución del proceso con el holón recurso, es decir, intercambia la información y los métodos acerca de cómo producir cierto producto utilizando cierto recurso. El conocimiento acerca de las posibles secuencias del proceso a ser ejecutado en el recurso, la estructura de datos que representan las salidas del proceso, métodos para acceder la información del diseño del proceso, etc.

También el holón producto intercambia conocimientos del proceso con el holón recurso, lo que es igual a, la información y los métodos con respecto a los progresos de la ejecución de los métodos en el recurso, el conocimiento acerca de cómo solicitar el arranque del proceso en los recursos, hacer reservaciones en el recurso, cómo monitorear el progreso

de la ejecución, cómo interrumpir un proceso, las consecuencias de interrumpir un proceso, suspender y reanudar el proceso en los recursos, etc.

De las relaciones anteriores se puede apreciar como los sistemas de producción basados en una holarquía se vuelven eficientes en el uso de los recursos y adaptables a los cambios en el ambiente en el cual existen ya que tienen holones especializados (OH, PH y RH) en cada parte del proceso intercambiando información y conocimiento, adaptándose a los cambios y manteniendo la independencia entre ellos.

HoMuCS

HoMuCS [6] es una arquitectura de sistemas, que se basa en la arquitectura de referencia HMS [9]. HoMuCS es el acrónimo para Holonic Multi-cell Control System (Control de Sistemas Holónicos Multicelulares).

Mientras que la arquitectura de referencia HMS sólo define una estructura general de un sistema de manufactura holónico, conformado por tres holones básicos, en HoMuCS se tienen cuatro, como se muestra en la figura 3.

En la figura se da una definición detallada, al puntualizar las operaciones y los atributos de cada clase representando los bloques básicos de la estructura. Las clases son descritas de tipo abstractas, usado para encapsular atributos y comportamientos que sean comunes a todos los holones de ese tipo.

La clase holón es la raíz de la heterarquía y prevé todos los servicios genéricos para un holón. Dentro de las características más importantes, tenemos el registro de un *Kernel*, el cual es el motor de un holón, éste maneja su lógica y, por consiguiente, es responsable del comportamiento autónomo y cooperativo. Adicionalmente la clase holón tiene servicios de identidad, lo que significa que todas las operaciones que le conciernen, son únicas del holón.

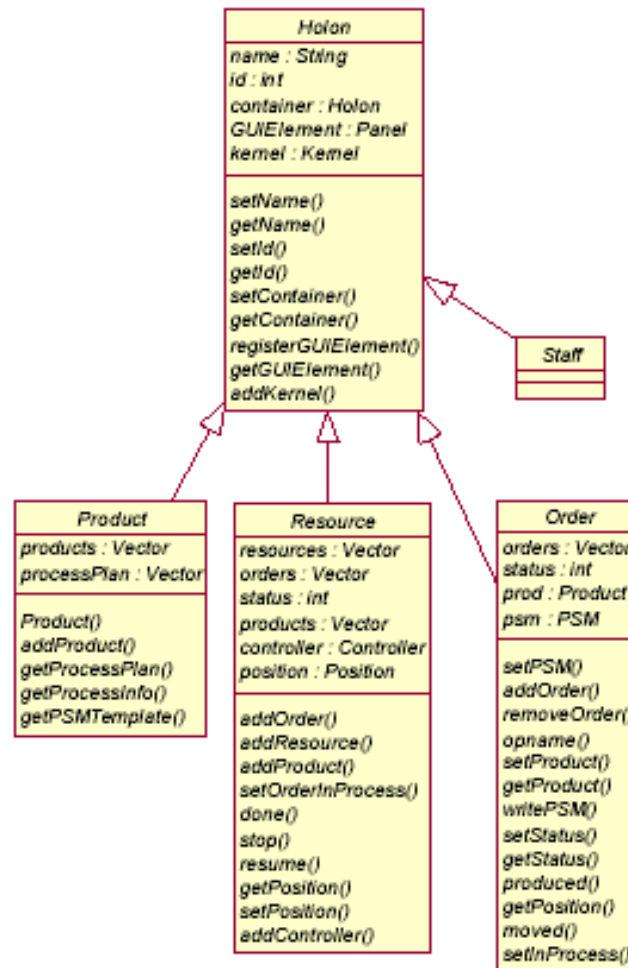


Figura 3: Las clases básicas de la arquitectura HoMuCS.

El holón producto representado por la clase producto es responsable de la generación y la entrega de información a los otros holones, por consiguiente éste tiene operaciones que ofrecen estos servicios, como la recuperación de los planes del proceso, información del proceso, etc.

El holón recurso es uno de los holones más complejos y la clase recurso encapsula varios atributos y operaciones que son genéricas para otros holones. Esta tiene atributos que contienen la lista de los recursos agregados, holones orden y producto dentro de ella y

operaciones para recuperar y agregar esos holones. Por otra parte, las operaciones “done”, “stop” y “resume” permiten el comportamiento de control de procesos. La clase recurso contiene el atributo “controller” que especifica la interfaz con los recursos físicos encapsulados dentro del holón. La operación “addController” se usa para registrar e inicializar la interfase de control con el recurso físico.

El holón Orden es representado por la clase Orden que encapsula algunas funciones genéricas necesarias para coordinar materiales y productos a través de nivel de la planta. Esta clase tiene un atributo para el estado de la información así como el atributo “orders” que se usa para la agregación de sub órdenes dentro de ésta. El atributo “psm” es un puntero hacia la información del modelo del estado del proceso. Las operaciones con respecto al estado de la información ofrecida están disponibles así como también las operaciones para manejar y adicionar información al modelo del estado del proceso (PSM). Las operaciones “produced” y “moved” permiten el proceso y manejo de las órdenes en HoMuCS.

Para finalizar, el Holón “Staff” representado por la clase “Staff” existe como una plantilla ya que es necesario asegurar una funcionalidad auxiliar, la cual se consigue con una especialización.

Coordinación / Optimización de Producciones Complejas

En resumen podríamos decir que tanto en HoMuCS como en la arquitectura HMS se pretende que los sistemas de producción de manufactura se vuelvan ágiles, cooperativos y sean capaces de adaptarse a los cambios en forma eficiente, para ello proponen holones (sistemas con partes físicas y partes de software de información) que se especialicen en cumplir con tareas específicas, trabajando de forma coordinada y cooperando entre ellos.

Ahora bien, en el artículo Coordinación/ Optimización de producciones complejas (COPC), se busca aplicar las teorías holónicas, pero esta vez, en los sistemas de *producción continua*. Para ello descomponen el modelo de producción en una agregación de pequeñas partes o unidades de producción, tal que, representen alguna parte física / química dentro de la producción.

Las unidades de producción son denotadas como PU y se consideran que son autónomas (al igual que los holones) porque están capacitadas para tomar sus propias decisiones por medio de su dispositivo de control local, también llamado supervisor, para mantener el sistema seguro y trabajando de acuerdo a las especificaciones recibidas de éste último.

Por otra parte, manejar los sistemas continuos de un proceso puede volverse una tarea sumamente compleja, lo cual resuelven transformando, a través de la aplicación de las teorías de sistemas a eventos discretos, el sistema continuo en eventos discretos, por ejemplo, mejor que saber a lo largo del tiempo la altura exacta del nivel de un tanque que se esta vaciando, es saber si esta vacío (evento1) o si no lo está (evento2), o saber cuando una válvula esta abierta o no, etc. Ésta transformación del sistema continuo en eventos discretos, indicará el estado actual de cada uno de los dispositivos involucrados en el proceso y al combinarlos, arrojará el estado actual del proceso.

Al hacer la transformación de la parte continua en eventos discretos, los estados posibles a ser manejados por el supervisor se simplifican y por lo tanto las decisiones que tomaran las unidades de producción autónomas, también.

La naturaleza de un sistema distribuido de producción exige una agregación de componentes que pueden ser controlados independientemente pero que también deben cooperar para lograr la meta de producción global, tal como se comportan los sistemas holónicos. Para ello proponen el modelo mostrado en la siguiente figura:

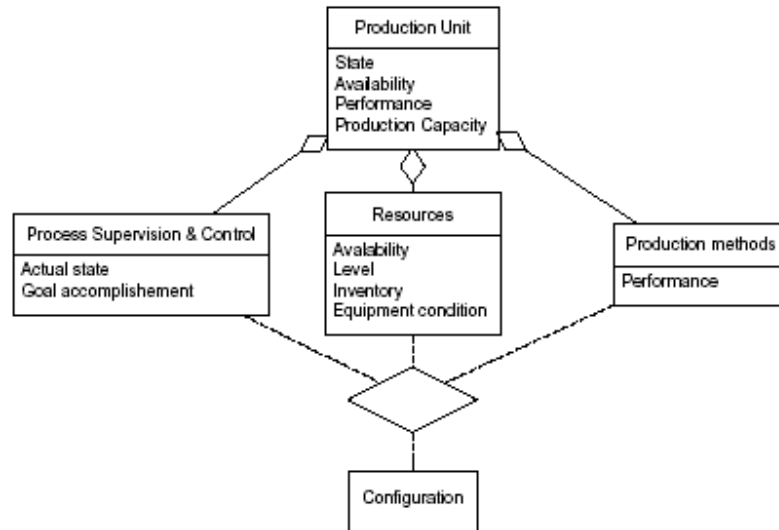


Figura 4: Modelo UML para una unidad de producción.

A cada unidad de producción se le agregan diferentes clases, que apoyan la consecución de sus metas, estas clases son: la clase Procesos de supervisión y control, la clase Recurso, la clase Métodos de Producción y por último la clase Configuración.

Clase Procesos de supervisión y control: esta clase controla la evolución del funcionamiento que se ejecuta actualmente en los recursos, según un método de producción para satisfacer la meta de producción. El elemento de control asociado a esta clase, observa la conducta y el estado del proceso, tanto como para transformar su control, así como también para informar a los operadores sobre fallas o errores en el proceso.

Clase Recurso: tiene toda la información sobre el estado de los recursos. Dentro de un sistema de producción se tienen diferentes procesos tales como: transformación, mezclado, transportación y procesos de almacenamiento, cada uno de ellos es realizado por equipos especializados, tales como, válvulas, bombas, tuberías, etc.

También existen otros tipos de recursos, como la energía y los servicios, los cuales son necesarios para realizar las actividades de la producción. Así que se les asocian algunos costos que también son manejados por esta clase. El consumo y el uso de la energía, orientan

las necesidades dentro del proceso de producción. Finalmente, algunas materias primas tienen que ser transformadas en productos intermedios o últimos. Así que el suministro y uso de materias primas también orientan las necesidades del proceso y tienen costos asociados. Los recursos pueden ser clasificados en los siguientes tipos:

- Equipos que transforman
 - a Operación de transformación
 - En la operación de transformación, varias entradas son transformadas en uno o más salidas. Tales operaciones se ejecutan en los sistemas de transformación como los reactores, y separadores.
 - b Operaciones de Almacenamiento
 - El papel de estas operaciones es guardar los productos. Los sistemas de almacenamiento son los vasos, los tanques, etc.,
 - c Operación de transporte
 - Los sistemas de transporte pueden mover los productos a lo largo de ciertas trayectorias y dentro de algunos límites.
- Las materias primas
- Los servicios
- d La energía.

Clase Métodos de Producción: tiene varias configuraciones posibles para las diferentes metas según las condiciones del recurso. El método de producción, tiene que ver con el tipo de producto a transformar, los estados actuales de los recursos y la configuración actual. En los sistemas de producción continua, se obtienen los productos a través de procesos que especifican los componentes y las referencias para las diferentes condiciones de operación. Los cambios en los componentes imponen los cambios en las condiciones de operación y viceversa.

Clase Configuración: define la configuración física de los equipos, los objetivos, el método de producción y el control de las referencias a ser usado. La configuración incluye: la

organización de recursos, la referencia de operaciones a ser logradas y su modo de operación (enlaces físicos de la infraestructura, los recursos, las referencias). La descripción de la configuración se obtiene de los métodos de producción a través de un sistema de decisión local. Esta clase describe qué recursos deben usarse y cómo deben organizarse para realizar todas las operaciones en una orden de producción. Cada operación requiere algunos recursos particulares, y deben producirse los productos intermedios para otras operaciones. Algunas operaciones pueden ejecutarse en paralelo, lo cual significa que deben organizarse los recursos físicamente para un lapso de tiempo. Cada nodo operacional tiene sus propias referencias y deben ser coherentes con los otros nodos de operación

Clase Unidad de producción: cada controlador de la PU recibe la información sobre la misión del proceso, objetivos y metas, regulación del proceso, y envía información al sistema externo para combinar y negociar las metas.

En los procesos de producción implementados a través de holones, es necesario tener varios niveles de coordinación jerárquica, para mantener los procesos controlados, desde el nivel superior táctico hasta el nivel técnico mas bajo. En [4] encontramos la siguiente figura:

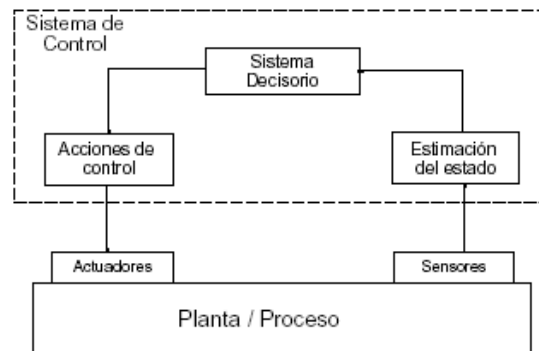


Figura 5: Componentes del sistema de control Jerárquico.

La información del nivel técnico más bajo (planta / proceso) es almacenada localmente, en sistemas orientados a control regulatorio. Estos equipos son construidos tolerantes a fallas, con un alto rendimiento en la ejecución de lazos simples de control. Los sistemas del segundo nivel deben mantener la comunicación con el operador, coordinar todos los procesos y servir

de interfaz con los sistemas administrativos. El tercer nivel está relacionado con la planificación a largo plazo así como con las interacciones con el ambiente externo a la empresa.

Una UP puede estar compuesta por dos sistemas: El físico que realiza los procesos y el lógico de decisión e información.

El sistema físico. está compuesto del equipo que realiza todas las actividades de producción físicas, usando las entradas de materia prima, energía, etc.

El sistema de decisión y de Información. Éste, es el sistema lógico que maneja la planta, usando la información y el conocimiento del proceso físico. También tiene la capacidad para establecer la comunicación con otras unidades de Decisión e Información que pertenezcan a otras UPs.

Las UPs llevan a cabo el proceso de producción de un tipo de producto. Realiza funciones de control y regulación y “conoce” cómo lograr sus metas de producción y el estado de su producción y pueden estar compuestas de una o más unidades de equipos que son las que realizan los procesos de transformación.

Las PUs deben saber sobre sus capacidades internas, estados, y restricciones. Pueden alinearse de manera tal que realicen diferentes procesos de producción o que cambien el nivel de producción basadas en los requisitos de la empresa o en su estado interno.

Sistemas Holónicos vs Sistemas Multi Agente

En primer lugar hay que destacar que un sistema compuesto de varios holones es llamado sistema holónico, e igual sucede con los agentes, un sistema compuesto de varios agentes es llamado sistema multi agente (MAS). Por otra parte, es común utilizar la palabra holón y/o agente de forma indiferente, lo cual crea confusión, así que en esta parte se intentará aclarar esto.

Al parecer las teorías de sistemas Holónicos y sistemas multi agente (MAS) están estrechamente relacionadas. Ambas discuten sistemas que consisten en objetos autónomos y

distribuidos. La primera diferencia es que los holones son más cooperativos que los agentes, tal como lo muestra la siguiente figura tomada de [7]. La segunda diferencia y quizá la principal es que los agentes sólo existen como software mientras que los holones son objetos físicos que consisten de ambos, de software y de entidades físicas.

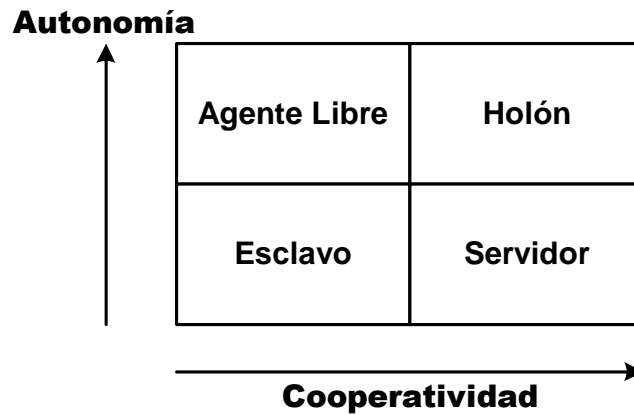


Figura 6: Comparación entre autonomía y cooperación para diferentes entidades

Los holones pueden por consiguiente caracterizarse como agentes físicos en un mundo físico como una antítesis de una red de computadoras [7]. Desde la perspectiva de MAS un sistema holónico es una especialización de un MAS, donde cada Holón es actualmente un agente. Sin embargo, los agentes de software en un HMS pueden usarse no solo como parte de un Holón sino también como entidades separadas que ayudan en el intercambio de información entre los holones, ayudando así en el flujo de información, apoyando la interacción entre holones, y aumentando el grado de reacción del sistema y su dinámica. Los agentes tienen dos aplicaciones potenciales principales en los sistemas holónicos de producción de manufactura y continua [3,7]. Primero las características inherentes de autonomía y cooperación de agentes pueden usarse para hacer a los holones autónomos y cooperativos. Segundo pueden usarse para aumentar la eficiencia del flujo de información e intercambio en el sistema, entre holones, y externamente con otros sistemas como las bases de datos, planificación de sistemas, etc.

*Capítulo II Bases Teóricas para Sistemas de Producción
Holónicos*

Existe una interdependencia entre los campos de MAS y HMS. Pero los MAS necesitan una infraestructura clara y bien definida, la cual debe incluir un modelo del sistema, protocolos estándares de interacción y especificaciones funcionales, por ello, en el próximo capítulo, se explicará lo concerniente a esto.

CAPÍTULO III

SISTEMAS MULTIAGENTES

Agentes

En el mundo de la informática, la tecnología de agentes ha aumentado su popularidad. El desarrollo continuo de la tecnología de red y la Internet han hecho que los agentes se vuelvan parte de casi cada sistema en la red. Esta nueva tecnología ofrece un mayor descubrimiento para los programadores que intentan resolver los problemas de comunicación y de aplicación. Sólo recientemente se ha tenido la idea de usar la teoría de agentes para los sistemas industriales. Los agentes son objetos de software autónomos que vagan dinámicamente en una red y algunas veces son inteligentes. Ellos tienen una visión local de su ambiente en la cual ellos también pueden reaccionar, mientras que el desempeño global de un sistema de agentes surge a través de su interacción dinámica con cada uno de los otros agentes. Ellos pueden ayudar a incrementar el grado de reacción de los sistemas industriales.

Hasta ahora la tecnología de agentes se ha usado principalmente en la informática. Sin embargo, los agentes parecen demostrar las habilidades de tecnología necesarias para la aplicación de nuevos conceptos industriales como HMS.

Todavía no hay una definición clara sobre qué es un agente, sin embargo el término agente es usado para describir una programación de software que puede transitar en una red, interactuar con "hosts", reunir información y regresar a su "casa". Si se considera un sistema de redes sin agentes, gran cantidad de datos tienen que ser transportados en la red, mientras que en un sistema con agentes, las llamadas a la red se reducen al mínimo. Además, los agentes, debido a su naturaleza, agregan flexibilidad y eficacia al sistema

En el capítulo anterior se comentó, que los sistemas Holónicos, tenían dos componentes principales, uno de software y otro físico. Se le podría atribuir al componente de software, la parte inteligente del mismo, es decir, el sistema que posee la información, que toma las decisiones, que tiene la capacidad para establecer comunicación con otros holonones, etc. Pues éste sistema lógico (software) puede ser implementado a través de Agentes. Al usar

“agentes inteligentes” que puedan controlar el sistema lógico se asegura que las decisiones tomadas por el holón, serán óptimas.

Características

Se podría decir que un Agente de software es un programa, que tiene una serie de rasgos o cualidades, que hacen que sea capaz de interactuar con su entorno, incluyendo a otros agentes de software y/o humanos, y aprender de él, tomando así sus propias decisiones basándose en el conocimiento acumulado. Para que esto suceda, el agente debe presentar, ciertas características, dentro de las cuales se encuentran las siguientes:

Creencias: es la noción que tiene un agente sobre sí mismo y sobre su entorno, quien es y de donde viene, quienes son sus iguales. Los deseos son los objetivos a largo plazo es decir, a donde va a llegar, la meta a cumplir. Las intenciones son los objetivos a corto plazo u objetivos actuales que le ayudan a cumplir con los objetivos principales. Los planes son los métodos para el cumplimiento de los objetivos principales y/o actuales.

Autónomo: es una de las características más importantes del concepto de agente. La idea de los agentes consiste en crear programas informáticos que tengan una serie de objetivos y posean unos conceptos del mundo, de tal forma que partiendo de sus conocimientos, los agentes sean capaces de aproximarse lo más posible a sus objetivos sin necesidad de que ningún usuario los guíe paso a paso hacia ellos.

Sociabilidad: cuando se trata con agentes no se piensa en una única entidad que se ejecuta de forma aislada es decir, los agentes son capaces de interactuar con otros agentes (humanos o no) a través de un lenguaje de comunicación entre agentes. Por lo tanto se suele pensar en sistemas complejos (multi-agente) en los que una serie de agentes colabora entre sí para llevar a cabo sus objetivos. Pueden comunicarse, además, con varios recursos del sistema o usuarios. Desde el punto de vista del agente los recursos pueden ser locales o remotos, hay un amplio rango de sistemas de recursos de agentes que pueden ser accedidos, por ejemplo programas de aplicación, bases de datos, sistemas de información, etc.

Capítulo III Sistemas Multi Agente

Reactividad: los agentes son capaces de percibir estímulos de su entorno (el mundo físico, un usuario vía una interfaz gráfica, una colección de otros agentes, Internet o todos ellos combinados) y reaccionar a dichos estímulos posiblemente para poder “conocer” en todo momento cómo es el “mundo” que le rodea

Proactividad, iniciativa: los agentes no sólo son entidades que reaccionan a un estímulo, sino que tienen un carácter emprendedor, y pueden actuar guiados por sus objetivos y decidir que acciones realizar para alcanzarlos. Es decir, un agente no sólo actúa en función de los estímulos, sino que puede realizar acciones que sean consecuencia de sus propias decisiones.

Movilidad: es la capacidad de un agente de trasladarse a través de una red telemática para desempeñar tareas específicas. Generalmente se puede identificar dos niveles de movilidad de los agentes:

Ejecución remota: un agente es transferido a un sistema remoto donde es activado y ejecutado en su totalidad, el mecanismo de transporte del agente utilizado varía desde TCP/IP a correo electrónico.

Migración: durante su ejecución un agente activo puede moverse de nodo a nodo para cumplir progresivamente su tarea, en otras palabras el agente puede suspender su ejecución, transportarse el mismo a otro nodo de la red y reanudar la ejecución desde el punto en el cual fue suspendida, además, pueden lanzar nuevos agentes durante su viaje, por ejemplo para adquirir información para su cliente o para ejecutar subtareas específicas;

Veracidad: asunción de que un agente no comunica información falsa a propósito;

Benevolencia: asunción de que un agente está dispuesto a ayudar a otros agentes si esto no entra en conflicto con sus propios objetivos;

Racionalidad: asunción de que un agente actúa de forma racional. Un agente se considera racional cuando tiene unos conocimientos de su entorno, unos objetivos deseables y unas reglas que determinan cómo alcanzar los objetivos a partir del

Capítulo III Sistemas Multi Agente

conocimiento que se tiene del medio intentando cumplir sus objetivos si son viables.

Cooperación: permiten la cooperación entre entidades de agentes, la complejidad de la cooperación puede variar desde un estilo de interacción cliente/servidor a negociaciones y cooperación basada en métodos de inteligencia artificial, tales como redes de contrato y protocolos. Esta cooperación puede necesitar del intercambio de información y representaciones de pre-requisitos para sistemas multiagentes.

Inteligencia: se refiere al método utilizado para desarrollar la lógica del agente o la inteligencia y está estrechamente relacionada con los lenguajes de agentes donde predominan dos aspectos: la creación de contenido pragmático del agente y la representación del conocimiento que proporciona los medios para expresar objetivos, tareas, preferencias y vocabulario apropiado para varios dominios. Este término es el que más problemas ha suscitado entre los autores ya que en definitiva, se promete mucho pero se entrega poco.

Adaptativo (Aprendizaje): para que un agente se adapte, debe de poder aprender a partir de las percepciones que recibe del entorno y a partir de sus conocimientos anteriores a la hora de resolver problemas. Si es capaz de aumentar su base de conocimientos de esta manera, podrá adaptarse a los entornos dinámicos en el tiempo, que son los del mundo real. Cambiando su comportamiento basado en las experiencias previas.

Operación asíncrona: el agente puede ejecutar tareas totalmente desacoplado de sus usuarios o de otros agentes, lo que significa que puede ser disparado por la ocurrencia de un evento particular.

Coherente. Todo el conocimiento del mundo que el agente tiene se almacena en una base de datos de conocimiento. Para que el comportamiento del agente sea el esperado, la organización del conocimiento debe ser coherente.

Muchas de las características anteriores, pueden garantizar que los agentes ayudan en la resolución del problema de la implementación de los sistemas Holónicos, en los procesos de producción, sin embargo, el agente no necesariamente debe de tenerlas todas, solo se necesita que cumpla con las dos principales (autonomía y cooperación) con esto es suficiente para considerarlo un agente y para garantizar que los holones serán autónomos con cierto grado de libertad para el cumplimiento del objetivo común principal.

La característica de la *Inteligencia*, ha causado mucha polémica en algunos autores que difieren en cuanto a este concepto. Si bien es cierto, que la autonomía es un rasgo esencial para distinguir a un agente de otro tipo de software, el hecho de que un agente sea autónomo, no necesariamente implica que sea inteligente, es decir, que las decisiones tomadas hayan sido en base al razonamiento y aprendizaje de los actos y sus consecuencias, pero aun así, en contradicción con lo anterior, se suele utilizar el concepto de autonomía para definir también al de inteligencia.

Parece evidente, que lo que se busca con el concepto de agente es añadir un nivel de abstracción a la hora de diseñar sistemas complejos. Se pretende modelar una entidad de software con las características propias del comportamiento social humano, de manera tal que la resolución del problema se acerque lo más posible a su resolución en el mundo real. Pero por ahora, la tecnología actual está lejos de permitir dicha utopía.

Los sistemas multi-agentes y el paradigma en el cual se han basado (“*paradigma de agentes*”) suponen una innovadora y prometedora manera de afrontar la resolución de problemas complejos. Se puede afirmar que los agentes representan un avance a la hora de afrontar la resolución de problemas que mediante el empleo de los paradigmas clásicos no habían obtenido una solución satisfactoria, como por ejemplo la rama de problemas centrados en sistemas de tiempo real. Estos sistemas necesitan un alto grado de flexibilidad y de adaptabilidad, debido a los constantes y rápidos cambios que sufre su entorno. Por ello, el paradigma de agente/sistemas multi-agente parece lo bastante apropiado para el desarrollo de sistemas en tiempo real estricto. Tal como se demuestra en [11] donde el principal obstáculo a nivel de control en tiempo real provenía de la dificultad de implementar los conceptos de sistemas multi agentes en un ambiente estocástico donde las limitaciones en tiempo real

debían ser encontradas para conseguir que el sistema de operación estuviera a salvo, es decir, el sistema, implementado a través de agentes, debía de reconfigurarse inteligentemente. Finalmente logran esta reconfiguración inteligente, gracias a la correcta implementación de los comportamientos de los agentes, adaptados al sistema en particular.

Después de la descripción expuesta anteriormente se puede deducir que el diseño de sistemas complejos con la tecnología de agentes resultaría conveniente de forma clara en al menos en los siguientes casos:

Sistemas distribuidos adaptables y con tolerancia a fallos que resuelven problemas complejos.

Sistemas abiertos en los que se añaden y eliminan componentes continuamente y en los que se necesita gran flexibilidad y escalabilidad.

Migración y distribución de la carga de procesamiento entre los nodos de la red.

Negociación entre distintos elementos del sistema para alcanzar soluciones (divide y vencerás).

Sistemas distribuidos en los que la personalización de los servicios prestados es un factor de gran importancia.

Estándares para los agentes

La tecnología de agentes se ha introducido en un amplio rango de aplicaciones, y se han desarrollado gran cantidad de sistemas de agentes. Los desarrolladores de todos estos sistemas no han tenido ninguna base a partir de la cual empezar, por tanto han utilizado soluciones diferentes para cada uno de los problemas encontrados. Todo esto ha provocado la pérdida de compatibilidad, portabilidad y reusabilidad. A parte, el desarrollo de estos sistemas se ha afrontado con un carácter particular, no cubriendo de esta forma todas las necesidades que se requieren en la industria para ser viable.

Por tanto, parece claro que se necesita la especificación de un estándar ampliamente aceptado por todas las partes que permita unificar los esfuerzos, y por lo tanto consiga que la

tecnología de agentes se convierta en una posibilidad real y no se quede en un montón de conceptos utilizados con objetivos meramente comerciales.

Organismos de estandarización.

Cuando se habla de esfuerzos en la estandarización de los Sistemas de Agentes son cuatro los organismos que sobresalen:

KSE

El *Knowledge Sharing Effort* (KSE) es un consorcio centrado en el desarrollo de normas que faciliten la colaboración y reutilización de bases de conocimiento y sistemas basados en el conocimiento. Algunos resultados de esta labor han sido las especificaciones del lenguaje KQML (Knowledge Querying and Manipulation Language) y del formato KIF (Knowledge Interchange Format), y la herramienta Ontolingua que permite la definición de ontologías.

El lenguaje KQML ha sido tomado como estándar de facto para el intercambio de conocimiento durante algún tiempo en los sistemas multi-agente.

No parece claro que el KSE siga todavía en activo aunque algunas de sus partes siguen trabajando como grupos de investigación independientes. Parece que nadie toma la responsabilidad de continuar con el desarrollo del estándar original.

The Agents Society

La Agents society fue creada con el espíritu de extender el desarrollo de sistemas de agentes y permitir que estas tecnologías de agentes inteligentes emergieran y tuvieran un lugar en el mercado.

Su labor es más de intercambio y recopilación de información sobre agentes, y parece que la actualización de su información no es todo lo ágil de lo que fuese deseable.

OMG

El OMG (Object Management Group) ha desarrollado el estándar MASIF (Mobile Agent System Interoperabilities Facility). En el desarrollo de este estándar han contribuido empresas como IBM, Crystaliz y GMD-Fokus. MASIF define interfaces para la transferencia y localización de agentes en sistemas de agentes implementados sobre CORBA. Es por tanto un estándar que facilita la creación de sistemas de agentes móviles.

FIPA

El estándar FIPA recoge todas las vistas que se tienen de un sistema de agentes (gestión, seguridad, movilidad, comunicación...) de tal forma que para cualquiera de esas vistas todos los esfuerzos se hagan en una única dirección. Éste estándar ha sido ampliamente reconocido internacionalmente y, por tanto, se ha convertido en una referencia a seguir en gran parte de los desarrollos basados en agentes.

Uno de los motivos de éste trabajo es el diseño de un sistema abierto, capaz de interactuar con otros sistemas de agentes heterogéneos por lo tanto, se buscó un mecanismo de comunicación que fuera compatible con la mayor parte de sistemas multiagentes de la actualidad. Por ello, se escogió el estándar más reconocido por la comunidad científica, como lo es FIPA.

FIPA

Descripción General de FIPA

La “Foundation for Intelligent Physical Agents” (FIPA) fue formada en 1996 para producir un software estándar para agentes heterogéneos y sistemas de agentes.

El progreso de estandarización empezó creando un consorcio internacional formado por organizaciones de distintas universidades y de la industria para de manera conjunta, desarrollar un estándar de carácter público. Este estándar se ocuparía de describir el comportamiento externo y las interfaces de los agentes y sistemas de agentes.

Declaración de intenciones de FIPA.

La actividad fundamental de los estándares de FIPA es facilitar la interoperabilidad de agentes y sistemas de agentes de diferentes vendedores. La misión principal es:

“La promoción de las tecnologías y las especificaciones que faciliten la colaboración entre sistemas de agentes comerciales heterogéneos dentro del comercio moderno y las especificaciones de la industria actual.”

Es importante destacar el énfasis que FIPA le da a la puesta en práctica de los sistemas de agentes dentro del comercio y la industria. De cualquier modo, las líneas de investigación se están centrando en la inteligencia o conocimiento de los agentes, lo que significa, software de sistemas que pueden tener potencial para razonar sobre ellos mismos y/o sobre otros sistemas que encuentre.

El mensaje central de FIPA es que a través de una combinación de actos de habla (“speech acts”), predicados lógicos y ontologías públicas se pueden ofrecer maneras estándar de comunicación entre agentes de forma que se entienda el significado de lo comunicado. Esto es mucho más ambicioso que, por ejemplo XML, que se limita a estandarizar la estructura semántica del documento.

Para mantenerlo, FIPA ha adoptado y está trabajando en especificaciones que parten de unas arquitecturas para sostener la comunicación entre agentes, lenguajes de comunicación y contenidos para expresar los mensajes y los protocolos de interacción que amplían el ámbito de un simple mensaje a una transacción completa. En el futuro hay planes para extender esto hasta copar una gran cantidad de formas de relación entre agentes.

Organización del estándar.

El estándar FIPA se compone de 2 partes: Referencias Tecnológicas y referencias a Aplicaciones.

La primera parte (normativa) se ocupa de especificar una plataforma tecnológica que permita el desarrollo de sistemas multi-agente abiertos.

Capítulo III Sistemas Multi Agente

La segunda parte (informativa) presenta algunas aplicaciones reales que motivan el uso de estos sistemas.

Referencias Tecnológicas.

Esta parte del estándar se ocupa de presentar las normas que deben regir el comportamiento e interfaces externas de un sistema de agentes y asegura la interoperabilidad con otros sistemas que cumplan dichas normas.

Por lo tanto el estándar no impone (aunque en algunas ocasiones da recomendaciones) como se debe implementar estos sistemas, sino que asegura que independientemente de las decisiones de diseño e implementación, van a ofrecer un comportamiento similar asegurando una total interoperabilidad.

FIPA se formó con el propósito de producir un estándar software para agentes heterogéneos y sistemas de agentes. Para ello está desarrollando el estándar FIPA, que recoge todas las vistas que se tienen hasta el momento de un sistema de agentes. De esta forma puede conseguir recoger todos los esfuerzos de investigación para dirigirlos en una única dirección.

En la realización de dicho estándar FIPA requiere la ayuda de sus miembros y de toda la comunidad que investiga en el campo de los agentes para conseguir una interoperabilidad total entre los sistemas de agentes desarrollados por las diferentes compañías y organizaciones.

El reconocimiento que ha tomado FIPA a nivel internacional, a nivel académico y en el ámbito industrial potencia las opciones de futuro no sólo de este estándar, sino de la tecnología de agentes en general.

Referencias a Aplicaciones (Informative Specification)

FIPA, consciente de que el futuro de los sistemas multi-agente pasa por su aplicación en la industria, ha introducido esta parte del estándar que pretende dar pistas de cómo introducir estos sistemas a nivel comercial.

Las partes que se incluyen son:

Part 4: Personal Travel Assistance.

Define un sistema de agentes que asiste al usuario al planificar un viaje, incluyendo la elección de itinerarios y medios de transporte, la reserva de billetes y hoteles, y ayuda durante el viaje dando información actualizada en cada momento.

Part 5: Personal Assistant.

Facilita las labores del usuario que se realizan de un modo regular y repetitivo durante su trabajo diario en la oficina, incluyendo la gestión del correo electrónico diario, las labores de administración del sistema, etc.

Part 6: Audio/Video Entertainment & Broadcasting.

Ayudan al usuario de sistemas digitales de entretenimiento (tanto sistemas broadcast como sistemas bajo demanda) a seleccionar entre los cientos e incluso miles de posibilidades que se le ofrecen. Los agentes de usuario colaborarían y negociarían con los agentes del proveedor del servicio para adecuar la programación a los intereses del cliente de acuerdo al perfil e intereses del mismo.

Part 7: Network Management & Provisioning.

Alrededor del mundo, numerosos proveedores de servicios de telecomunicaciones ofrecen servicios que se componen de elementos pertenecientes a diferentes operadores. El objetivo es ofrecer en cada momento las mejores condiciones del servicio en cuanto a calidad y coste. El sistema que se presenta permite una gestión descentralizada de redes de estos servicios, aplicada principalmente a las Redes Privadas Virtuales (VPNs).

Part 9: Product Design & Manufacturing.

Presenta una aplicación que recoge el proceso de fabricación de un producto en todas sus fases. Pretende utilizar la mayor parte del estándar: Ontologías,

Interacción con el usuario, Seguridad y Movilidad. Aparece con la versión FIPA 98.

El lenguaje de comunicación de los agentes.

Introducción a los lenguajes de comunicación de agentes.

Para resolver la comunicación entre agentes se utilizará el lenguaje ACL siguiendo de esta forma las especificaciones propuestas por FIPA.

FIPA-ACL (al igual que el KQML) es un metalenguaje, es decir, un lenguaje utilizado para comunicar contenidos de otros lenguajes, independientes de los agentes.

Es cierto que FIPA se basó en KQML para desarrollar FIPA-ACL, lo cual los convierte en lenguajes muy similares. Ambos se basan en actos comunicativos, de tal forma que el significado completo se consigue gracias al sentido que el acto comunicativo imprime al contenido del mensaje. También los lenguajes comparten una sintaxis similar.

En términos comparativos, FIPA-ACL proporciona una mejor estructuración del mensaje así como mayor formalismo en las especificaciones de los actos comunicativos que KQML. También es más simple, puesto que solo se centra en los mensajes de contenido, separándolos de los mensajes que ayudan a gestionar la sociedad de agentes. Estos últimos se incluyen en una especificación separada (Gestión de Agentes).

Tampoco se proveen mensajes relacionados con el transporte, como sería el del servicio de difusión. Por último, los servicios que no entren dentro de los mensajes de contenido, o dentro de otras especificaciones de FIPA se dejan a cargo del lenguaje de contenido, como puede ser la gestión de múltiples respuestas.

FIPA-ACL, en líneas generales es más completo que KQML, pero, la principal razón por la que se usa es por su mayor difusión y por su espíritu estándar.

El lenguaje ACL.

El elemento básico de la comunicación entre agentes es el mensaje. En las siguientes secciones se comprobarán las especificaciones que ha realizado FIPA para determinar cual es la sintaxis que ha de tener un mensaje y su utilización.

Cada mensaje tiene una estructura prefijada que está formada de una serie de parámetros. La utilización de estos parámetros está sujeta a las necesidades requeridas por el mensaje. Los parámetros o campos (*slots*) se han clasificado en diferentes categorías según su función. En la siguiente tabla se describen cuales son y a que categoría pertenecen:

Tabla 1: Parámetros del mensaje.

Nombre del parámetro	Categoría del elemento
Sender	Participante de la comunicación
Receiver	Participante de la comunicación
Content	Contenido del mensaje
Encoding	Descripción del contenido
Protocol	Control de conversación
reply_with	Control de conversación
reply-by	Control de conversación
Performative	Definición de tipo de mensaje
reply_to	Participante de la comunicación
Language	Descripción del contenido

Ontology	Descripción del contenido
conversation-id	Control de conversación
in-reply_to	Control de conversación

Definición del tipo de mensaje que se envía

Dentro de esta categoría sólo se tiene un elemento, performative la cual indica el tipo de acto de comunicación (communicative act) que se quiere hacer constar en el mensaje ACL. Se tienen diferentes tipos de actos de comunicación:

Tabla 2: Tipos de actos de comunicación.

Actos de Comunicación	Descripción
accept-proposal	Aceptación de una propuesta para realizar una Acción
agree.	Indica que se está de acuerdo con la realización de una acción
Cancel	Cancelación de una petición anterior que aun no ha finalizado.
cfp (Call For Proposal)	Pedir propuestas a un conjunto de agentes para realizar una acción
Confirm	Se confirma al receptor que una proposición es verdadera.
Disconfirm	Expresa lo contrario que confirm. Se envía para indicar que una proposición es falsa cuando se creía que era cierta Verdadera.
Failure	Se informa del fallo de una petición hecha con Anterioridad.
Inform	Se informa al receptor del mensaje de algún hecho.
inform-if	Se informa al receptor si una proposición es cierta o no.

inform-ref	Se informa al receptor si un objeto al cual se hace referencia es cierto o no.
not-understood	El mensaje no ha sido entendido.
Propagate	Un agente envía un mensaje junto con un conjunto de AIDs a los cuales se reenviará el mensaje. El agente que lo recibe lo interpreta y lo vuelve a enviar. Hace falta una restricción con el fin de cortar la cadena de retransmisiones del mensaje, como por ejemplo un timeout.
Proxy	El agente que envía el mensaje (sender) espera que el agente que reciba la petición identifique a los agentes que satisfacen una descripción dada y les reenvíe el mensaje.
query-if	Se formula una pregunta a cerca de una proposición.
query-ref	Se formula una pregunta.
Refuse	Se rechaza una petición.
reject-proposal	Es el contrario de accept-proposal
Request	Se pide al agente la realización de alguna acción
request-when	Se pide al agente la realización de alguna acción bajo ciertas condiciones.
request-whenever	Se pide al agente la realización de alguna acción cuando se cumplan unas condiciones determinadas.
propose	Realización de una propuesta como contestación de un cfp previo.

Subscribe	Suscribe a un agente para la realización de un servicio específico.
-----------	---

Participantes de la comunicación

Un participante de la comunicación es un identificador de un agente, y puede estar situado en alguno de los siguientes 3 campos del mensaje:

sender: Identifica el agente que envía el mensaje.

receiver: Identifica el agente que ha de recibir el mensaje. Es un conjunto de agentes con uno o más identificadores (*multicast*).

reply-to: Identificación del agente al que tiene que ir destinado el siguiente mensaje de la conversación. Puede ser el mismo agente que ha enviado el mensaje (que aparecerá en el campo sender) o no, pudiéndose crear de esta forma conversación de más de dos agentes.

La identificación de un agente tiene un formato fijo que cabe tener en cuenta:

AID: (*agent-identifier*: ***nameagente*** @ *jade/amo/1088* : *addresses* (***sequence*** *üop*://
juan @ jade/amo/1088))

:sender <*AID*>

:receiver (*set* (*agent-identifier* <*AID*>) (*agent-identifier* <*AID*>))

El identificador del agente o AID (Agente Identifier), es una tupla formada por 2 elementos, el nombre que se le ha dado al agente y la dirección física de la máquina donde reside utilizando para ello el protocolo IIOP. El nombre siempre debe de ir seguido de la máquina donde se encuentra.

Contenido del mensaje

A esta categoría sólo pertenece el parámetro: *content*.

El contenido del mensaje puede tener diferentes formatos según las necesidades que se tengan. FIPA define cuatro tipos predefinidos: FIPA-CCL, FIPA-SL, FIPA-KIF y FIPA-RDF. No hace falta reseñar que si uno quiere se puede crear su propio lenguaje con su propia semántica (*plain text*), pero en este se pierde toda la compatibilidad que aportan las especificaciones señaladas anteriormente.

El lenguaje FIPA-CCL (FIPA Constraint Choice Language) define una semántica que permite especificar predicados que satisfacen restricciones (CSP).

Soporta la representación de problemas, recolección de información, técnicas de fusión y la aplicación de técnicas de resolución de problemas.

El lenguaje FIPA-SL (FIPA Semantic Language) permite formar expresiones lógicas, intercambiar de forma óptima conocimiento ente agentes, y expresar acciones a realizar. Esta pensado de manera general y puede ser aplicado en multitud de dominios diferentes.

El lenguaje FIPA-KIF (FIPA Knowledge Interchange Format) permite expresar objetos como términos y proposiciones como sentencias. No permite expresar acciones.

El lenguaje FIPA-RDF (FIPA Resource Description Framework) es muy parecido al FIPA-SL y permite representar objetos, interacciones y acciones entre ellos.

Descripción del contenido

En la sección anterior se ha descrito los formatos que puede poseer el campo: *content*. Cuando un agente recibe un mensaje este debe poder identificar ¿qué se le está enviando? y ¿cual es el formato?, para este efecto se proporcionan una serie de campos. FIPA define tres descriptores de contenido con la misma función esencial, pero que definen cosas diferentes.

language:

Define el lenguaje con el cual se ha escrito el contenido. Los posibles valores de este campo pueden ser FIPA-CCL, FIPA-SL, FIPA-RDF, plain, Prolog, Java, Lisp, etc. El valor de este campo puede ser redefinido completamente por cualquier agente. Cuando un agente recibe un mensaje, mirando este campo sabrá si podrá analizarlo o no, ya que él mismo sabe sus propias posibilidades. Por ejemplo si sólo puede analizar mensajes en LISP y le llega uno en Prolog, el agente responderá inmediatamente con la performativa *not-understood*.

encoding:

El contenido del mensaje puede codificarse de muchas formas, aunque normalmente se transmiten cadenas de caracteres, y es ésta especificación la que se describe en este campo. Es opcional, en caso de estar vacía, se hará servir la codificación que se utilice para interpretar el mensaje entero.

ontology:

Este campo es muy importante. Define la ontología utilizada para darle significado al contenido del mensaje. Por ejemplo, se puede recibir un mensaje que sea (es (rojo)). Este mensaje, aparentemente sin sentido, debemos saber qué significa, de qué se nos está hablando. Así, para este ejemplo,

podría haber definida una ontología nombrada semáforo donde rojo significará el color del semáforo. Y la palabra “es” sería aplicada a una acción cierta. De esta forma el agente que recibiera el mensaje anterior podría deducir fácilmente que un semáforo está en rojo.

Control de conversación.

protocol: Identifica el tipo de protocolo de interacción dentro del cual se envía el mensaje.

conversation_id: Es un identificador de una conversación entera. Una conversación puede implicar el envío y la recepción de varios mensajes, en cuyo caso todos los mensajes tendrán la misma identificación.

reply_with: En este campo se almacenará un identificador utilizado para seguir los diferentes puntos de una conversación, es decir, un identificador que indicará a qué punto de la conversación pertenece este mensaje.

in_reply_to: Este campo almacenará a qué mensaje de los recibidos anteriormente se está contestando con el mensaje actual. Cuando un agente *i* envía a un agente *j* un mensaje en el que el contenido del campo: *reply-with* es *expr* el agente *j* al responder al agente *i* llenará el campo: *in-reply-to* con *expr*, y el agente *j* en el momento lo reciba continuará con la conversación.

reply_by: Es una marca de Timeout. Aquí se especifica una fecha y una hora que indica la fecha de caducidad del mensaje. A partir de este momento el que ha recibido el mensaje podrá hacerle caso o descartarlo directamente dependiendo de la implementación.

FIPA-ACL es el lenguaje más utilizado en la mayoría de implementaciones de plataformas y es el recomendado por el estándar.

Protocolos de interacción FIPA

A continuación se expondrán los diferentes Protocolos de Interacción que se encuentran recogidos en el estándar de FIPA. Dichos Protocolos de Interacción son secuencias de mensajes que aparecen frecuentemente durante las comunicaciones entre los agentes. La utilización de dichos Protocolos en la elaboración de agentes asegura que un agente, mientras siga estos protocolos, no llegará a una conversación sin sentido.

En una conversación normal entre agentes a menudo suelen ocurrir ciertas secuencias de mensajes, donde en un punto de la conversación ciertos mensajes son enviados y otros se esperan recibir.

Cabe destacar, que para FIPA, una conversación es una comunicación entre dos Agentes, y un diálogo es una comunicación particular dentro de una conversación. Es decir, por cada conversación podemos tener muchos diálogos y un diálogo pertenece a una única conversación. Los diálogos se pueden entender como una instancia de un protocolo de interacción de FIPA.

A estas secuencias típicas de mensajes es lo que se le suele llamar *Protocolo de Interacción (IP)*. Existen 2 posibles opciones de implementación:

Cada agente es el encargado de enviar y recibir espontáneamente una serie de mensajes, con lo cual la implementación de dicho agente se hace muy complicada.

El agente sigue una serie de Protocolos de Interacción en cuyo caso esta garantizado que no puede llegar a conversaciones sin significado, lo que hace mucho más fácil la implementación del agente.

FIPA da una serie de protocolos de Interacción generales para la interacción entre agentes a los que se les otorga un nombre predefinido. Esto no quiere decir que necesariamente se tengan que implementar todos o que no se puedan desarrollar e implementar nuevos, pero si emplea uno de los que hay predefinidos, el protocolo deberá seguir la especificación de FIPA.

Capítulo III Sistemas Multi Agente

Por su naturaleza, un agente puede mantener múltiples diálogos simultáneamente. Se llama conversación a cada diálogo particular, es decir, un agente puede concurrentemente mantener varias conversaciones con varios agentes y utilizar varios protocolos. La especificación de los siguientes protocolos se da para una conversación particular.

Aunque como ya se ha señalado un agente FIPA no necesita implementar ninguno de los IP de FIPA, excepto aquellos requeridos para el manejo del Agente.

Con la recolección de las definiciones de IP se pretende el uso de protocolos de interacción estándares en agentes desarrollados en diferentes contextos.

Los criterios mínimos que debe tener un protocolo para cumplir las reglas FIPA son:

Una representación clara del protocolo usando un diagrama AUML

Una clara y sustanciosa documentación

La utilización del nuevo protocolo debe quedar clara.

FIPA Request Interaction Protocol

El protocolo Request Interaction de FIPA simplemente permite a un agente pedir a otro la realización de una acción. El agente que recibe la petición puede realizar la acción o responder, de algún modo, que no la ha hecho.

A continuación se muestra la representación de este IP:

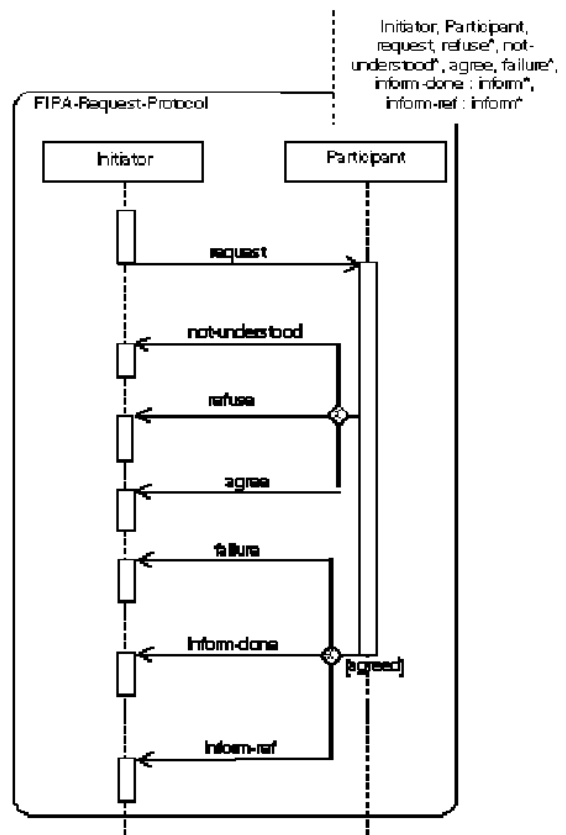


Figura 7. Protocolo FIPA – Request

FIPA Query Interaction Protocol

En el protocolo “Query Interaction” de FIPA, se le pide al receptor la realización de algún tipo de acción de tipo “*inform*”. La petición de un *inform* es una pregunta, y hay 2 actos para realizarla: *query-if* o *query-ref*.

Cualquiera de estos 2 puede ser utilizado para iniciar este protocolo. En cualquier caso, un *inform* es utilizado como respuesta, aunque el contenido del *inform* dado como respuesta en el caso de un *query-ref* es una expresión con una referencia al mensaje inicial.

A continuación se muestra la representación de este IP:

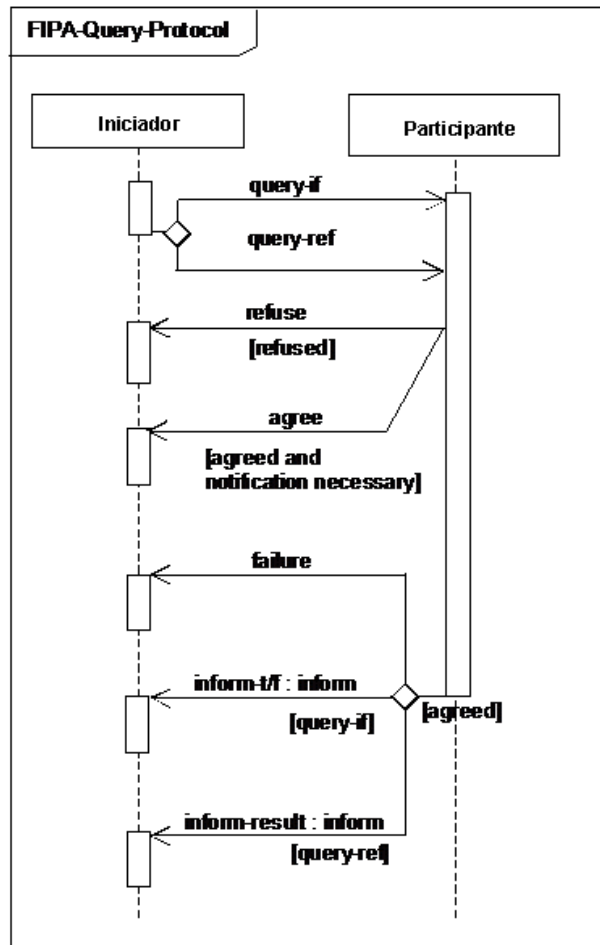


Figura 8: Protocolo FIPA – Query.

FIPA Request When Interaction Protocol

El Protocolo de interacción “Request When” de FIPA proporciona un marco para el acto de comunicación *request-when*. El iniciador utiliza una acción *request-when* para pedir al receptor la realización de alguna acción en el momento en el que una condición sea cierta. Si el agente receptor entiende la petición e inicialmente no la rechaza, enviará *agree* y esperará hasta que la precondition ocurra. Entonces intentará realizar la acción y notificará el resultado al que ha realizado la petición. Si después del acuerdo inicial el receptor no puede realizar la acción, enviará un *refuse* al emisor.

A continuación se muestra la representación de este IP:

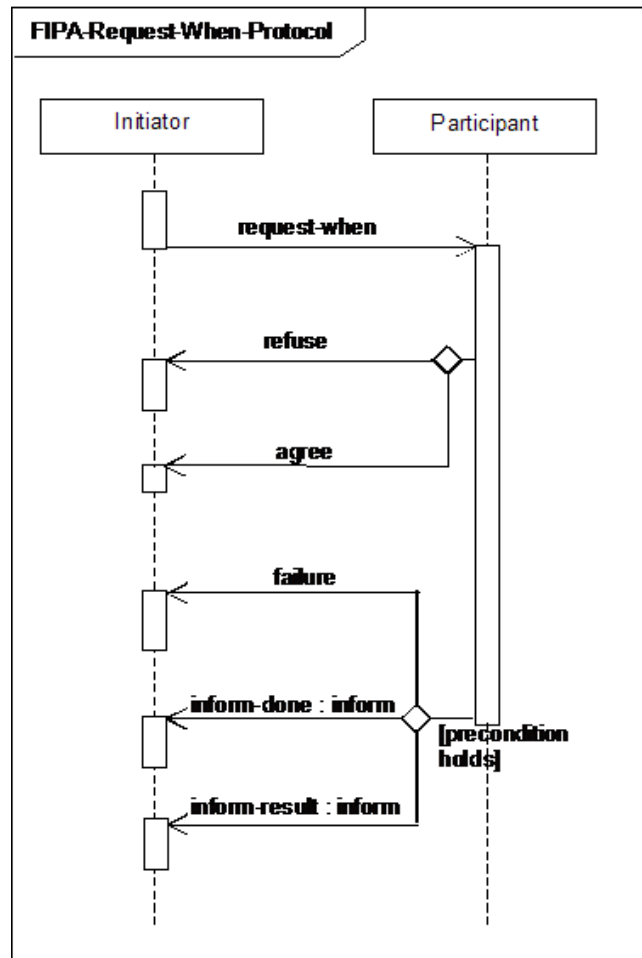


Figura 9. Protocolo FIPA Request When

FIPA Contract Net Interaction Protocol

Esta especificación presenta una versión del protocolo Contract uno de los más usados, originalmente desarrollado por Smith y Davis. Este Protocolo es una modificación mínima del modelo original en el cual se añade actos de comunicación de rechazo y de confirmación. En el Contract Net, el agente que toma el papel de jefe quiere que uno o más agentes realicen algún tipo de tarea optimizando una función que caracteriza dicha tarea. Esta característica es expresada comúnmente como el precio, pero podría ser el tiempo de finalización más corto, la distribución óptima de las tareas, etc.

El “jefe” solicita que otros agentes realicen propuestas para la realización de la tarea mediante un acto de comunicación *call for proposals (cfp)*, donde además de especificar la tarea a realizar puede adjuntar cualquier condición que crea necesaria a la hora de la realización de la tarea. Los agentes que reciben dicha llamada son vistos como contratistas potenciales y son capaces de generar propuestas para la realización de las tareas mediante un acto de comunicación *propose*. La propuesta del contratista incluye las precondiciones que este tiene para realizar la tarea, como puede ser el precio, el tiempo en que se acabará el trabajo, etc. Alternativamente, el contratista puede negarse mediante un *refuse* a realizar propuestas.

Una vez el tiempo de espera se ha cumplido, el “jefe” evalúa las propuestas recibidas y selecciona el agente que realizará la acción. Puede seleccionar uno, varios o ninguno. A los agentes que han realizado las propuestas seleccionadas, si las hay, se les enviará un acto de comunicación *accept-proposal* y los otros recibirán un acto de comunicación *reject-proposal*. Las propuestas están unidas al contratista, así que una vez el jefe acepta una propuesta, el contratista adquiere el compromiso de realizar la tarea. Cuando el contratista ha completado la tarea, envía un mensaje al “jefe” señalando que la tarea se ha completado.

Hay que darse cuenta que este IP requiere que el “jefe” sepa cuando se han recibido todas las respuestas. En el caso en que un contratista no pueda contestar ni con un *propose* ni con un *refuse*, el jefe podría quedarse esperando indefinidamente. Para evitar esto, la llamada *call for proposal* incluye un tiempo de espera que señalará qué respuestas serán recibidas por el jefe. Las propuestas recibidas después del tiempo de espera serán automáticamente rechazadas alegando que la propuesta llegó tarde.

A continuación se muestra la representación de este IP:

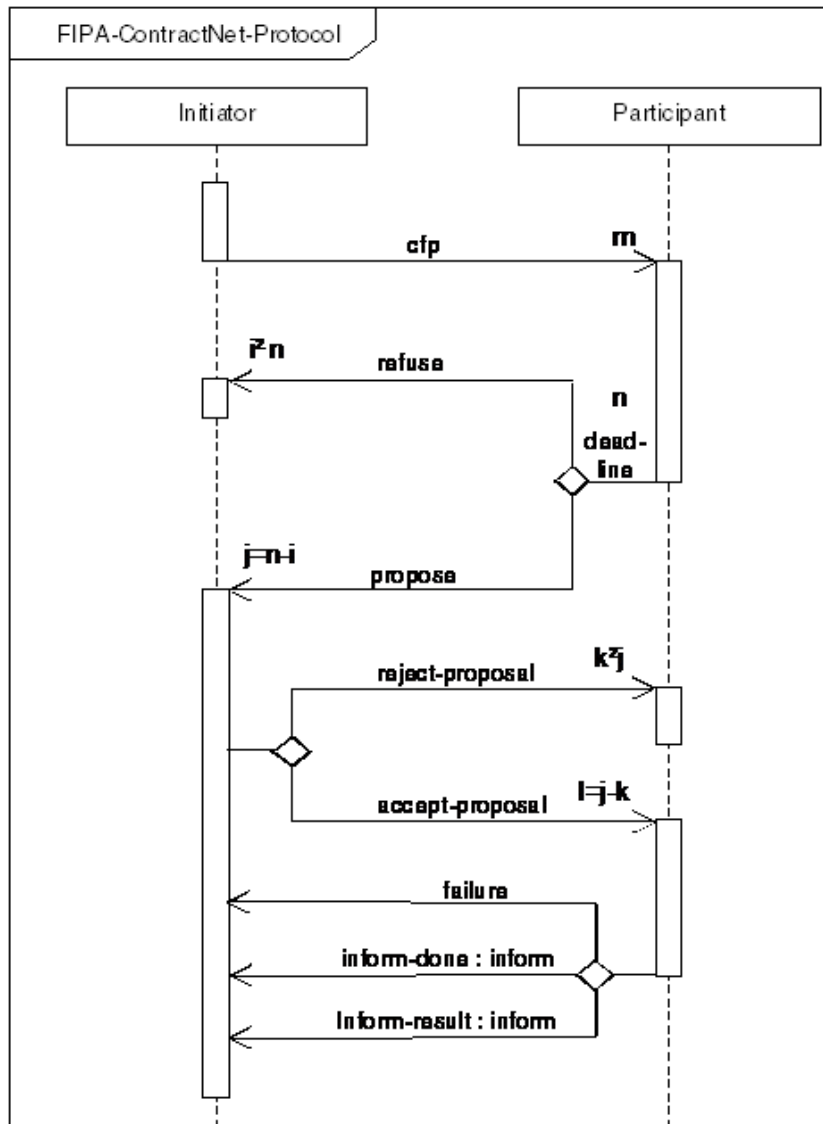


Figura 10. Protocolo FIPA Contract Net

FIPA Iterated Contract Net Interaction Protocol

El Protocolo de Interacción Iterated Contract Net de FIPA es una extensión del protocolo básico de CONTRACT NET, pero se diferencia en permitir hacer varias rondas iterativas de ofertas. Como en FIPA Contract Net, el “jefe” realiza la llamada para la obtención de propuestas con un acto de comunicación *cfp*. Los contratistas entonces contestan con sus ofertas con un *propose* y el “jefe” puede entonces aceptar una o más de las

ofertas, rechazarlas todas o puede iterar en el proceso lanzando un *cfp* revisado. La intención es que el “jefe” busque conseguir mejores ofertas de los contratistas modificando la llamada y haciendo nuevas peticiones (a partir de una revisión de las anteriores). El proceso termina cuando el “jefe” rechaza todas las propuestas y no realiza ningún nuevo *cfp*, acepta una o más ofertas o los contratistas se niegan a realizar nuevas ofertas.

A continuación se muestra la representación de este IP:

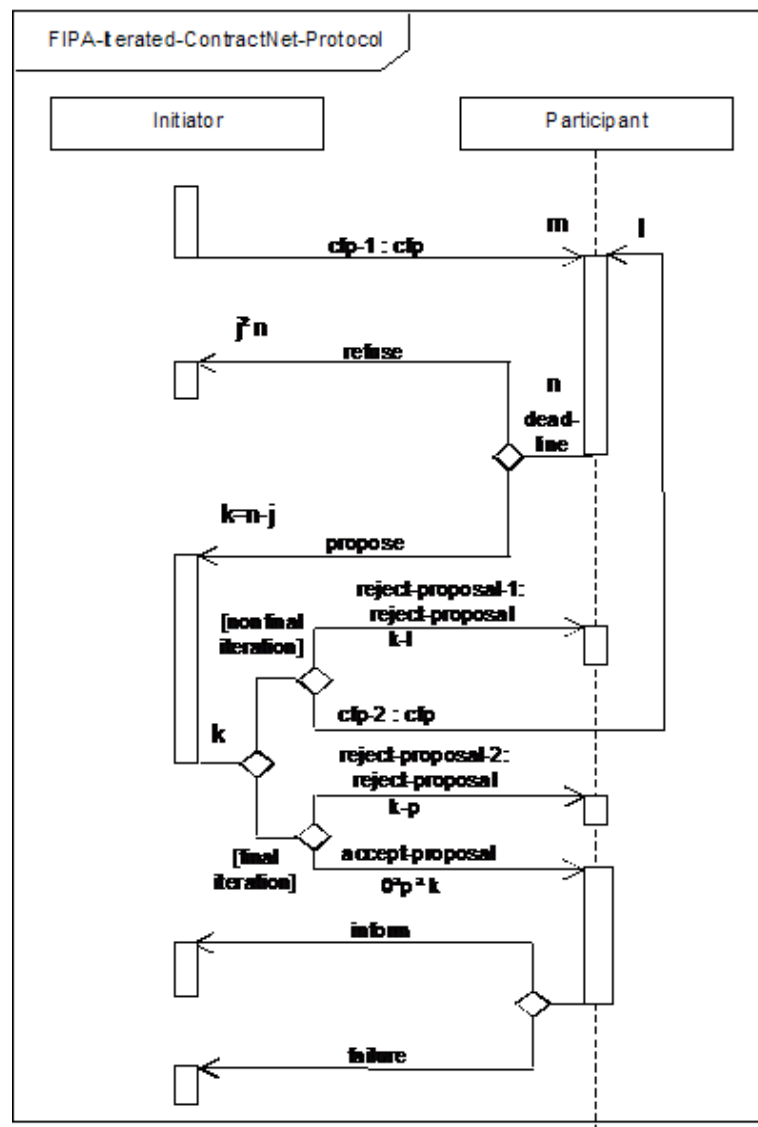


Figura 11. Protocolo FIPA Iterated Contract Net.

FIPA English Auction Interaction Protocol

En el FIPA English Auction Interaction Protocol, el subastador busca el precio de mercado de una mercancía proponiendo inicialmente un precio por debajo del que se supone que tiene, para ir subiendo gradualmente el precio. Cada vez que se anuncia el precio, el subastador espera para ver si algún comprador esta dispuesto a pagar el precio propuesto. Tan pronto como un comprador indica que acepta el precio, el subastador realiza una nueva llamada con un precio incrementado para recibir ofertas. La subasta continúa hasta que no haya compradores dispuestos a pagar el precio propuesto, momento en el que la subasta acaba.

Si el último precio que fue aceptado por un comprador excede el precio de reserva del subastador (que es privado), la mercancía es vendida a este comprador por el precio acordado, sino la mercancía no se vende.

En la figura, el subastador llama, mediante un *cfp* que genera un multicast a todos los participantes en la subasta. Hay que hacer notar que en una subasta física, la presencia de los participantes en una única habitación conlleva que la aceptación de una oferta se transmite simultáneamente a todos los participantes, no sólo al subastador. Esto puede no ser cierto en un mercado de agentes, en cuyo caso es posible que varios agentes hagan ofertas a un mismo precio. La subasta continuará mientras se reciba aunque sea una sola oferta con lo que los agentes necesitan saber cuando su oferta (representada mediante un acto de comunicación *propose*) ha sido aceptada. De ahí la aparición en el IP del *acceptproposal* y *reject-proposal*, a pesar de que este comportamiento es implícito en el modelo real de la Subasta en Inglés.

Destacar que las propuestas que son enviadas por los oferentes sobre todo son concernientes al proceso de ofertas. Como respuesta a un *cfp* para comprar una mercancía, una propuesta sería algo de la forma: “Yo aseguro que estoy dispuesto a pagar Z por X, propongo que la oferta sobrepase dicho valor.” Esto permite al subastador estar seguro que los oferentes están dispuestos a pagar el precio sin comprometerse a pagarlo actualmente hasta que el subastador pida específicamente el precio propuesto por la oferta ganadora.

Al final del IP, el subastador típicamente manda un *request* IP con la oferta ganadora para completar la transacción de la subasta.

A continuación se muestra la representación de este IP:

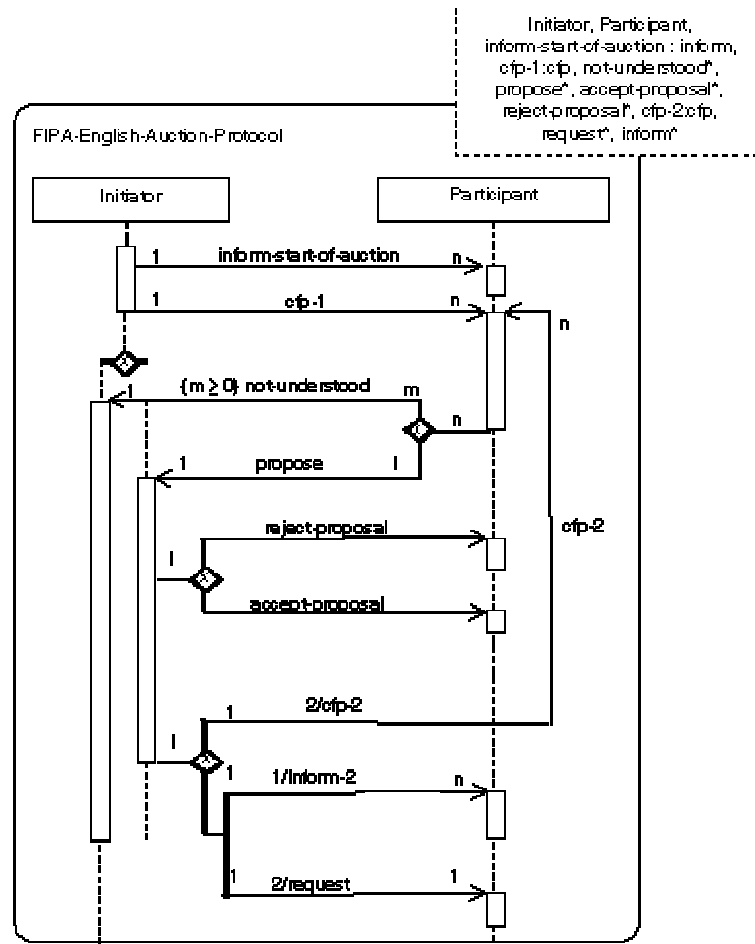


Figura 12. Protocolo FIPA English Auction

FIPA Dutch Auction Interaction Protocol

En el protocolo Dutch Auction Interaction Protocol de FIPA, el subastador intenta encontrar el precio de mercado de una mercancía empezando por una oferta de un precio mucho mayor que el precio esperado del mercado, para ir reduciendo el precio paulatinamente hasta que un comprador lo acepte. La tasa de reducción del precio la fija el

subastador que, además, normalmente se reserva un precio del que no baja. Si la subasta reduce el precio por debajo del precio de reserva entonces la subasta se termina.

El término “Dutch Auction” deriva del mercado de flores de Holanda, donde es el método dominante a la hora de encontrar el valor de mercado de las cosas, típicamente las flores. En el modelado de la subasta de flores Holandesas real, algunas complejidades adicionales ocurren:

Primero, la mercancía puede ser dividida. Así por ejemplo, el subastador puede estar vendiendo 5 cajas de tulipanes a un precio X , y un comprador puede comprarle sólo 3 cajas. La subasta entonces continua, con un precio en el siguiente paso por debajo de X , hasta que el resto de la mercancía es vendida o el precio de reserva es alcanzado. Esta venta parcial de la mercancía esta presente sólo en algunos mercados, en otros el comprador debe de hacer una oferta para comprar la mercancía entera.

Segundo, el mecanismo del mercado de flores esta hecho para asegurar que no existen contiendas entre los compradores previniendo ninguna otra oferta una vez se ha hecho. Las ofertas y licitaciones están comprometidas, así no hay ningún protocolo para aceptar o rechazar una licitación.

En el caso de los agentes es demasiado restrictivo requerir que dichas condiciones se apliquen, así, es bastante posible que 2 o más ofertas sean recibidas por el subastador para la misma mercancía, de ahí que el protocolo expuesto permita rechazar una oferta. Esto esta destinado sólo para el caso de múltiples competidores y ofertas simultáneas. Queda fuera del ámbito de esta especificación el especificar algún mecanismo particular para resolver dichos conflictos.

A continuación se muestra la representación de este IP:

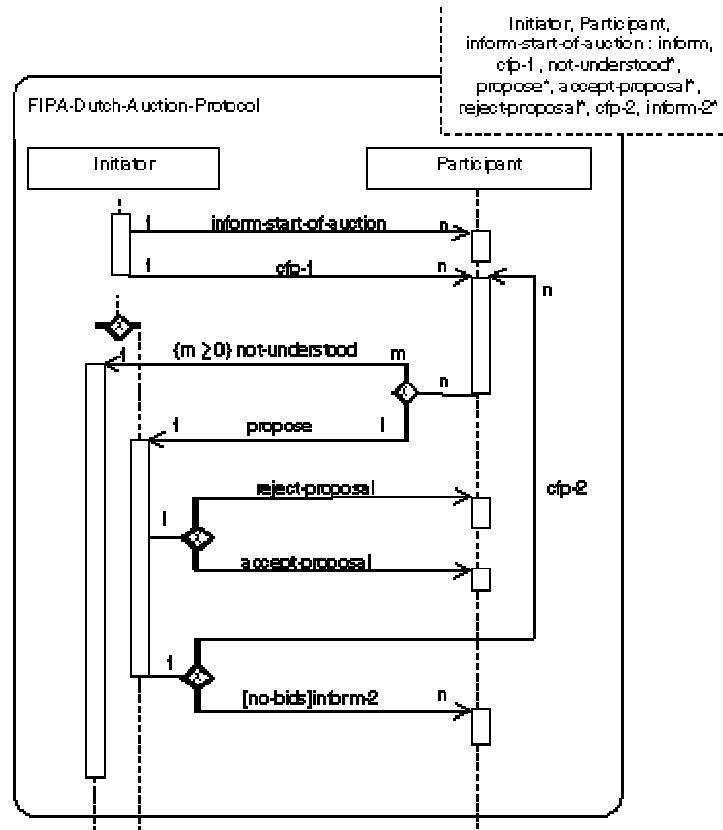


Figura 13. Protocolo FIPA Dutch Auction

FIPA Brokering Interaction Protocol

El concepto de un brokerage de información ha sido ampliamente usado en sistemas mediados en general y particularmente en sistemas multiagentes. El Protocolo de FIPA Brokering Interaction Protocol (IP) está diseñado para soportar estas interacciones de brokerage en los sistemas multiagentes.

En general, un broker es un agente que ofrece un conjunto de servicios de comunicación a otros agentes usando algún tipo de conocimiento acerca de las capacidades y necesidades de estos agentes. Un ejemplo típico de brokering podría ser el de un agente que pregunta a un broker sobre uno o más agentes que puedan contestar una pregunta. El broker entonces determina un conjunto de agentes apropiados a los que enviar la pregunta, les envía la pregunta y entrega la respuesta al agente que hizo la petición. El uso de agentes brokers

puede simplificar de una forma significativa la labor de interaccionar con agentes en un sistema multiagente. Además los agentes brokers también permiten que un sistema sea adaptable y robusto en situaciones dinámicas, soportando controles de escalabilidad y seguridad implementándolos en dichos agentes.

El FIPA Brokering IP es una macro IP, porque el acto de comunicación *proxy* lleva encapsulado un acto de comunicación como su argumento y así el IP para este acto también irá encapsulado en este IP. Cuando el acto de comunicación encapsulado incluye alguna acción que podría ser realizada por los agentes determinados por el broker, este IP podría ser extendido para notificar el resultado de las acciones.

El agente broker podría grabar algunos parámetros ACL, como por ejemplo:

:conversation-id, *:reply-with* y *:sender*, del mensaje *proxy* recibido para poder enviar las respuestas al agente original correspondiente (el que ha enviado el mensaje de *proxy*).

A continuación se muestra la representación de este IP:

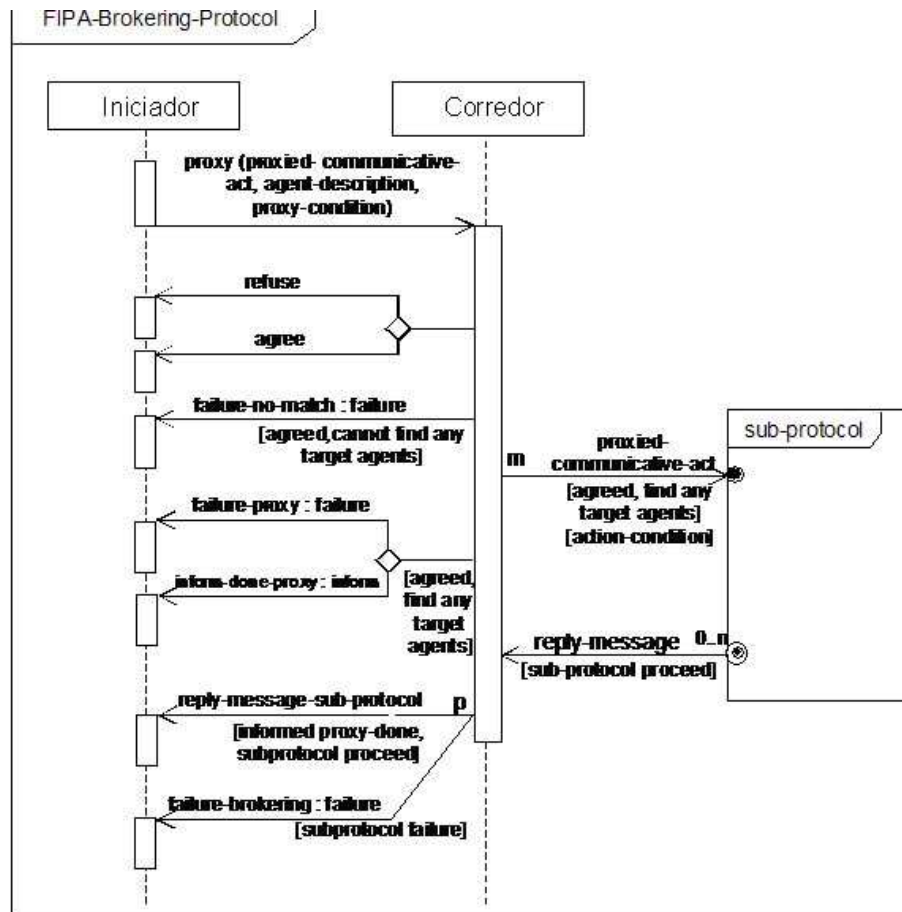


Figura 14: Protocolo FIPA Brokering

FIPA Recruiting Interaction Protocol

El concepto de un “brokerage” de información ha sido ampliamente usado en sistemas mediados en general y en particular en sistemas multiagentes. El Protocolo Recruiting Interaction Protocol de FIPA esta diseñado para soportar estas interacciones de brokerage en los sistemas multiagentes.

Hablando generalmente, un “broker” es un agente que ofrece un conjunto de servicios de comunicación a otros agentes usando algún tipo de conocimiento acerca de las capacidades y necesidades de estos agentes. Un ejemplo típico de brokering podría ser el de un agente que pregunta a un broker sobre uno o más agentes que puedan contestar una

pregunta. El broker entonces determina un conjunto de agentes apropiados a los que enviar la pregunta, les envía la pregunta y entrega la respuesta al agente que hizo la petición. El uso de agentes brokers puede simplificar de una forma significativa la labor de interactuar con los agentes en un sistema multiagente. Además, los agentes brokers también permiten que un sistema sea adaptable y robusto en situaciones dinámicas, soportando controles escalabilidad y seguridad en el agente broker.

La principal diferencia entre este protocolo y el anterior es que las respuestas de los agentes destino seleccionados van directamente al iniciador de la petición o a algún receptor que ha sido designado. El uso de agentes brokers puede simplificar de una forma significativa la labor de interactuar con agentes en un sistema multiagente.

Además dichos agentes también permiten que un sistema sea adaptable y robusto en situaciones dinámicas, soportando controles escalabilidad y seguridad en el agente broker.

El FIPA Recruiting IP es una macro IP, porque el acto de comunicación *proxy* lleva encapsulado un acto de comunicación como su argumento y así el IP para este acto también irá encapsulado en este IP. Cuando el acto de comunicación encapsulado incluye alguna acción que podría ser realizada por los agentes determinados por el broker, este IP podría ser extendido para notificar el resultado de las acciones.

A continuación se muestra la representación de este IP:

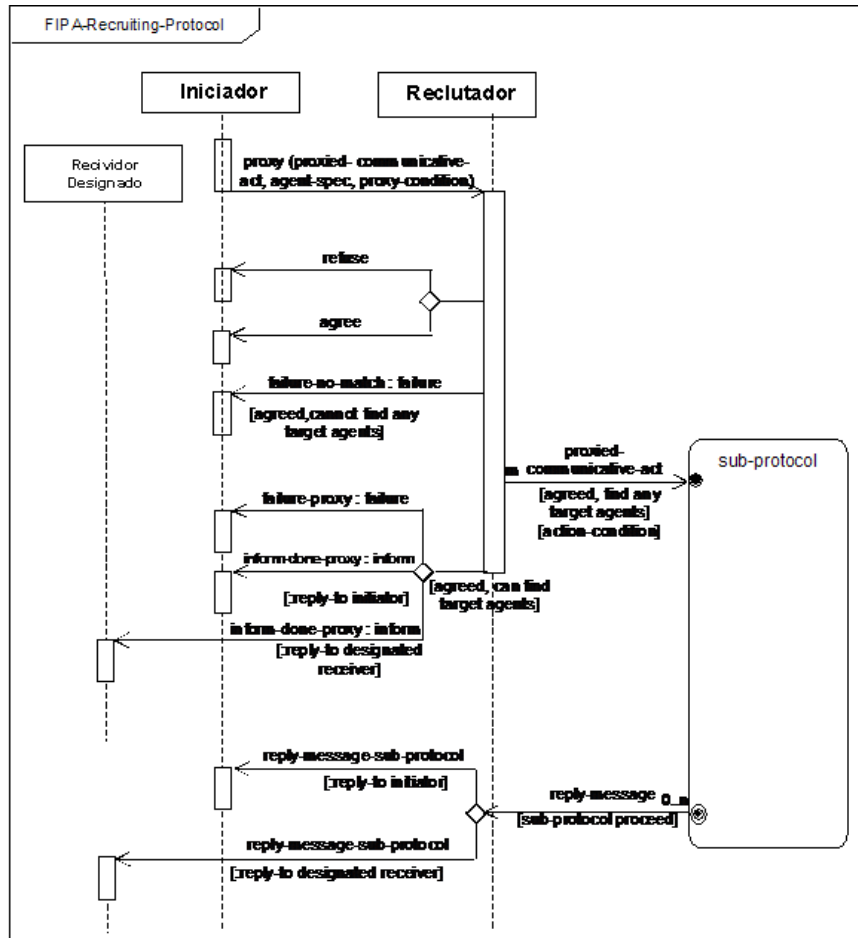


Figura 15: Protocolo FIPA Recruiting

FIPA Propose Interaction Protocol

En el FIPA Propose Interaction Protocol (IP), un agente iniciador propone a los agentes receptores que él hará las acciones descritas en el acto de comunicación *propose* cuando el agente receptor acepte esta propuesta. Al completar este IP con un *acceptproposal* típicamente conllevará la realización de la acción propuesta y el retorno de una respuesta de estado.

A continuación se muestra la representación de este IP:

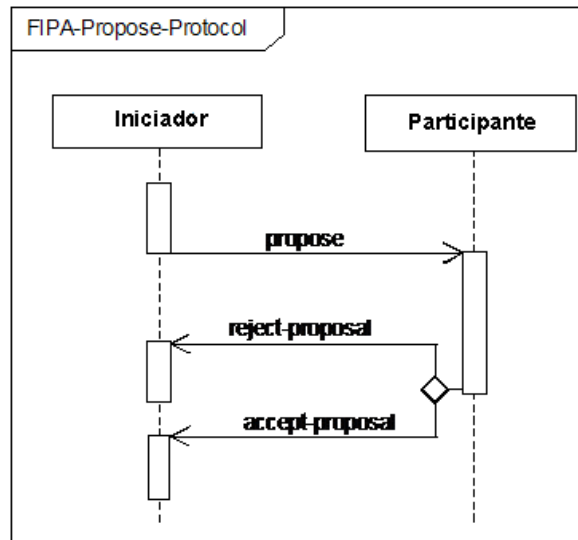


Figura 16: Protocolo FIPA Propose

Los protocolos expuestos anteriormente, son los que FIPA proporciona para la interacción entre agentes. Un sistema, para ser *FIPA-compliant* no necesita implementar todos y cada uno de los protocolos expuestos.

Herramientas para la implementación de agentes

Como se dijo anteriormente, en este trabajo, se pretende el diseño de un sistema abierto, capaz de interactuar con otros sistemas de agentes heterogéneos, por ello, tomando en cuenta las sugerencias de FIPA, de los entornos para programación de agentes, se decidió escoger a JADE, éste se basa en el lenguaje de programación java, al igual que casi todos, los entornos de programación de agentes, recomendados por FIPA.

Java, es un lenguaje orientado a objetos, y fue escogido debido a su gran popularidad, fuertes nociones de seguridad (C++ no sería un lenguaje conveniente por razones de seguridad), el garbage collector (reciclador de memoria dinámica), así como por su soporte multi hilo y multi plataforma. Los lenguajes como C o C++, deben ser compilados para un chip, y si éste se cambia, todo el software debe compilarse de nuevo, lo cual encarece mucho los desarrollos [33].

Capítulo III Sistemas Multi Agente

Dentro de las desventajas que Java presenta es que no ha tenido una gran historia de uso en tiempo real debido a su gran tamaño y su comportamiento no determinístico, entre otras cosas. Sin embargo recientemente se ha conseguido considerables progresos en el desarrollo de microprocesadores basados en Java y “kernels” de Java en tiempo real [11, 34,35], por lo tanto, es de esperar que en corto tiempo, el uso que tiene Java en tiempo real, aumente.

JADE

JADE (*Java Agent Development Framework*) es un marco de software cuya meta es simplificar el desarrollo de los sistemas de multi-agente asegurando la conformidad con el estándar a través de un sistema comprensivo de servicios y de agentes de sistema en conformidad con las especificaciones de FIPA: servicio de páginas amarillas, transporte de mensajes y analizadores de servicios, y una biblioteca de los protocolos de interacción FIPA lista para ser utilizada.

La plataforma de agentes de JADE está en conformidad con las especificaciones del estándar FIPA, ya que incluye todos los componentes obligatorios del control de la plataforma; estos son el acumulador, el AMS, y el DF.

Toda la comunicación entre agentes se realiza a través de los mensajes que se envían, todos ellos escritos mediante FIPA ACL.

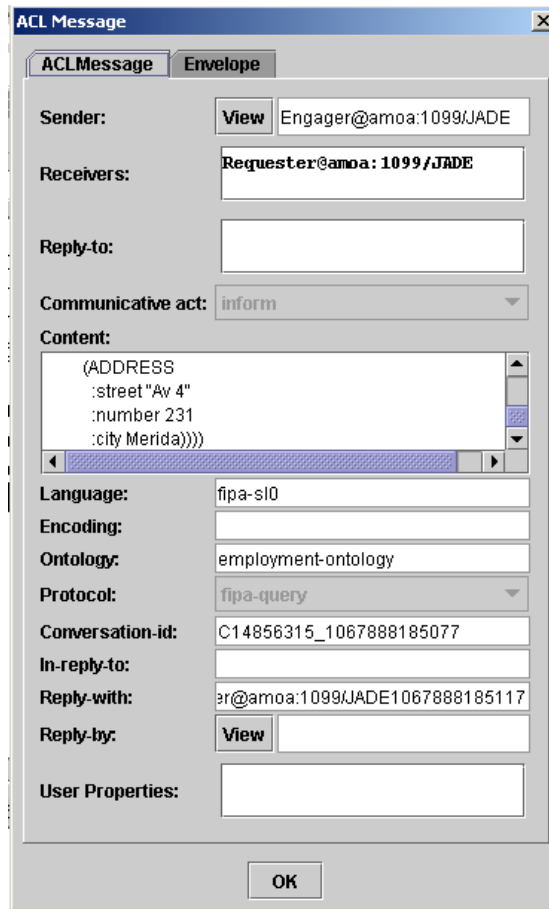


Figura 17. Ventana en JADE de un mensaje ACL

La plataforma de agentes se puede distribuir en varios anfitriones (containers), pero solo una máquina virtual de Java (JVM) se ejecuta en cada anfitrión. Cada JVM es básicamente un envase de agentes que proporciona un tiempo de ejecución (runtime) completo para la ejecución del agente y permite que varios agentes se ejecuten concurrentemente en el mismo anfitrión.

La arquitectura de la comunicación ofrece una mensajería flexible y eficiente, donde JADE crea y maneja una cola de mensajes ACL entrantes, privada a cada agente; los agentes pueden tener acceso a su cola, a través de una combinación de varios modos: *el bloqueo, la interrogación, el descanso*. Se ha puesto en ejecución el modelo completo de la comunicación de FIPA y sus componentes se han integrado completamente: protocolos de

interacción, sobre ACL, esquemas de codificación, ontologías y, finalmente, protocolos de transporte. El mecanismo de transporte se adapta a cada situación de forma transparente eligiendo el mejor protocolo disponible. Java RMI, HTTP, e IIOP son los utilizados actualmente, pero se pueden agregar más protocolos fácilmente vía las interfaces de JADE (MTP y el IMTP).

La mayoría de los protocolos de interacción definidos por FIPA están ya disponibles y pueden ser “instanciados” después de definir el comportamiento de cada estado del protocolo. El SL y la ontología de la gerencia de los agentes están disponibles, así como la ayuda para los idiomas y las ontologías definidas por el usuario.

Básicamente, los agentes se ponen en ejecución mediante un hilo de ejecución cíclico. Siguiendo la solución multi-hilos, ofrecida directamente por el lenguaje JAVA, JADE favorece también la programación de los comportamientos cooperativos, dejando que estos se programen de una manera ligera y eficaz. Se incluyen una serie de comportamientos programados y listos para utilizar que resuelven las tareas más comunes de los agentes, por ejemplo protocolos de la interacción de FIPA. Estos comportamientos tienen un conjunto de reglas que les permiten despertarse bajo ciertas condiciones, etc.

La plataforma del agente proporciona un GUI para la gerencia externa, supervisión y control del estado de los agentes, permitiendo, por ejemplo, parar y reiniciar agentes. El GUI permite también crear e iniciar la ejecución de un agente en un anfitrión externo, a condición de que un contenedor de agentes esté funcionando previamente. El GUI permite también controlar otras plataformas FIPA externas al agente.

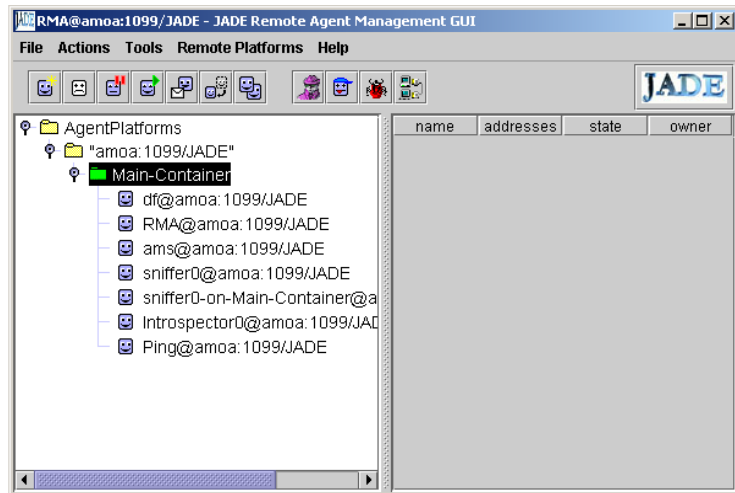


Figura 18. Ventana en JADE del agente manejador remoto (RMA).

La GUI del DF (agente de páginas amarillas) se puede lanzar desde el menú de las herramientas del RMA. Usando esta GUI, el usuario puede trabajar con el DF: con las descripciones de los agentes, el registro y los agentes registrados. Puede modificar las descripciones de los agentes registrados, y también puede buscar las descripciones de ciertos agentes. El GUI permite también agrupar el DF con otros DF's y crear una red compleja de dominios primarios y secundarios de servicios de Páginas Amarillas.

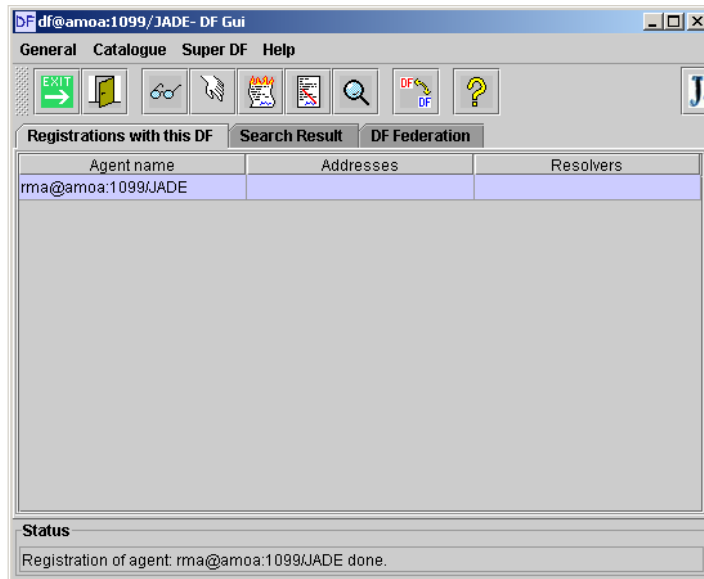


Figura 19. Ventana en JADE del agente de páginas amarillas (DF).

Agentes del sistema.

Sobre la base de JADE se han puesto en ejecución un número de herramientas gráficas o agentes de sistema que apoyan la fase de depuración y detección de errores que, generalmente, es muy compleja en los sistemas distribuidos, uno de ellos es el siguiente:

DumyAgent es una herramienta simple, pero de una extrema utilidad para examinar los intercambios de mensajes entre agentes. Facilita la validación de la interfaz del agente antes de su integración en el AMS, así como la prueba en caso que un agente esté fallando. La interfaz gráfica proporciona la ayuda para corregir, para componer y para enviar mensajes ACL a los agentes, para recibir de los agentes, y, eventualmente, para cargar o guardar algún mensaje en disco.

Este agente, es de mucha utilidad para comprobar el correcto funcionamiento de los desarrollos en la plataforma.

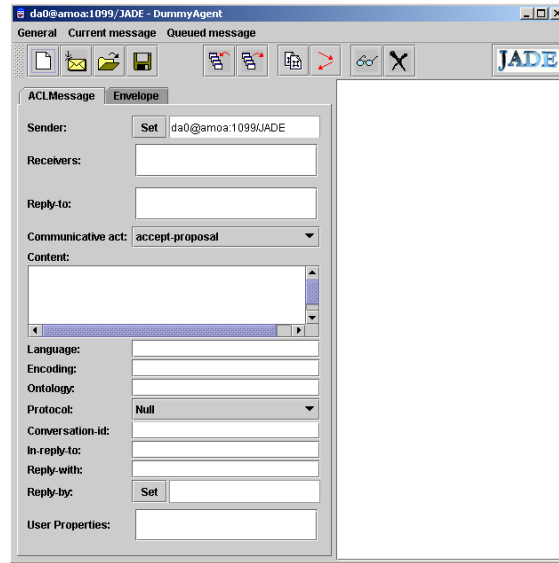


Figura 20. Ventana en JADE del Agente Dumy

El agente Sniffer permite seguir los mensajes intercambiados en una plataforma de agentes JADE. Cuando el usuario decide activarlo sobre un agente, o a un grupo de agentes, cada mensaje que reciben o envían se sigue y se muestra en la ventana del Sniffer. El usuario puede seguir la pista de todos los mensajes e incluso acceder a su contenido.

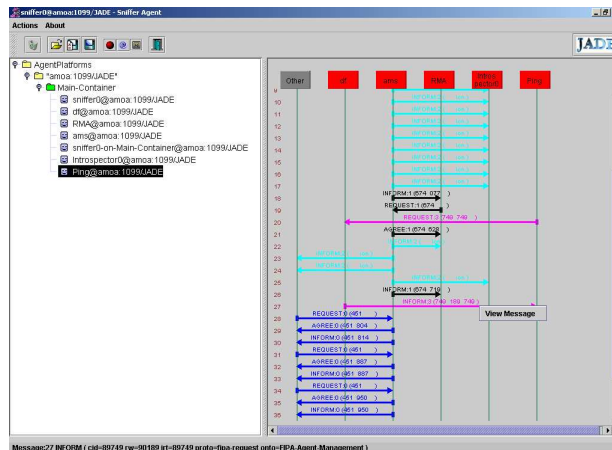


Figura 21. Ventana en JADE del Agente Sniffer.

Capítulo III Sistemas Multi Agente

El agente *Introspector* es otro de los agentes facilitado por la plataforma, éste agente permite supervisar y controlar el ciclo vital de un agente en ejecución y de sus mensajes, permitiendo ver la cola de mensajes enviados y recibidos.

Por último, sería bueno decir que JADE es una marca registrada por CSELT y es principalmente el resultado de una dura actividad de investigación. Ha participado con éxito en las pruebas de la interoperabilidad de FIPA y se ha utilizado extensivamente en el marco de varios proyectos internacionales.

Todavía se continúa el desarrollo de JADE. Mejoras, realces, y nuevas pruebas ya se han planeado, la mayoría de ellas en colaboración con los usuarios interesados de la comunidad de JADE.

Jade (diferencias entre hilos y comportamientos)

MEJORAR LA EXPLICACIÓN DE JADE; YA QUE LA QUE ESTÁ ACÄ ES MUY ESCUETA.

CAPÍTULO IV

METODOLOGIA Y DISEÑO

En este capítulo se desea mostrar todo lo que implica el diseño de implementar un sistema holónico en un sistema de producción continua, para eso se ha seguido la metodología recomendada en el artículo COPC, explicada anteriormente en capítulo II. Aún así, a lo largo de este capítulo se explicará en forma más detallada la integración de ésta metodología a las otras herramientas y su aplicación al sistema de producción continua escogido: la extracción de petróleo en un grupo de pozos.

Como se ha dicho en diferentes oportunidades, se desea que el sistema holónico sea ágil, adaptable, autónomo, cooperante, etc., para ello se utiliza el concepto de Unidad de Producción en representación del holón. La misma ésta conformada por dos capas: la parte física y la parte de decisión e información, la primera lleva a cabo lo referente al proceso físico (modelaje, observación de variables, control regulatorio, etc.); la segunda capa, de decisión e información, se implementa a través de MAS [3, 4, 5, 7, 11], explicado en el capítulo III, con ello se logra llevar a cabo un control supervisorio de la parte física, y tomar decisiones en base a la teoría holónica, debido a las grandes ventajas que representa tener a los sistemas multiagentes. La implementación y programación de los agentes se lleva a cabo, gracias a la plataforma de JADE, como una de las plataformas recomendadas por FIPA. Allí se utiliza a java como lenguaje de programación, lo que proporciona grandes ventajas por ser un lenguaje multiplataformas orientado a objetos.

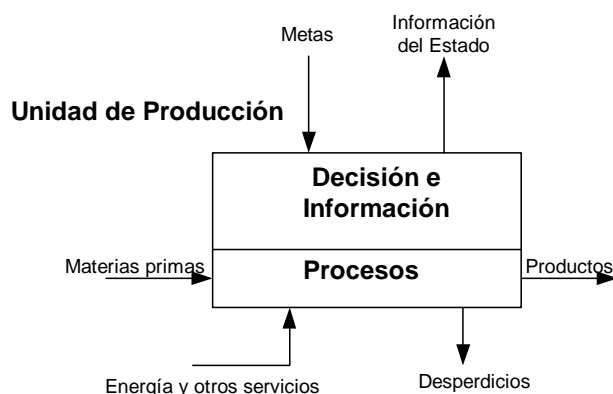


Figura 22: Descripción de la Unidad de Producción

Vista general sobre la extracción del petróleo crudo de un pozo

El petróleo es una mezcla de hidrocarburos compuestos que contienen en su estructura molecular carbono e hidrógeno principalmente. El número de átomos de carbono y la forma en que están colocados dentro de las moléculas de los diferentes compuestos proporciona al petróleo diferentes propiedades físicas y químicas. Se puede describir como un líquido viscoso cuyo color varía entre amarillo, pardo oscuro y negro. Además tiene un olor muy característico y flota en el agua. Se encuentra en un inmenso bloque de roca que yace muy profundo en el subsuelo a lo cual se le llama yacimiento o reservorio.

Para su extracción se perfora la tierra hasta la roca donde se encuentra el petróleo contenido (reservorio o yacimiento), creándose así, un pozo. Una vez que el pozo es perforado, se bajan sensores dentro del pozo por un cable para obtener una imagen detallada del yacimiento. Estos sensores responden una amplia gama de preguntas a lo largo de la vida del yacimiento. La primera pregunta importante es: ¿cuál es la distribución exacta de los fluidos dentro de la capa que contiene petróleo?

La respuesta se puede obtener midiendo la presión del yacimiento. Así que se introduce un dispositivo de prueba en el hoyo por medio de un cable. Se detiene en la primera estación (un punto de referencia escogido previamente) y se presiona contra la pared del pozo. Se introduce una sonda dentro de la formación y se toma una muestra de la presión de los fluidos en la roca. Se retira luego la sonda y se levanta hasta el siguiente punto de muestra. Y de

nuevo, se mide la presión a esa profundidad. A medida que este procedimiento se repite, se dibuja una curva que muestra los cambios que sufre la presión con la profundidad.

Los codos en la curva de presión indican los cambios de la clase de fluido en la roca. Esto ayuda a saber, dentro de un rango de pocos metros, el límite vertical entre el gas, el petróleo y el agua, información crítica que permite extraer la mayor cantidad de petróleo por pozo. Algunas mediciones adicionales de la presión y el flujo pueden indicar todavía más:

- La rapidez con que el yacimiento producirá
- El volumen de reservas que contiene
- La distancia del pozo a los límites del yacimiento.

El fluido que se encuentra en el yacimiento, no está formado solo por petróleo, también contiene, gas, agua, lodo, piedras, etc.

Una vez que se ha hecho una evaluación completa, el pozo está en condiciones de producir, es decir, de trasladar continuamente el fluido desde el yacimiento hasta la superficie. Algunas veces, el fluido asciende en forma natural, dependiendo de algunas circunstancias como la profundidad del yacimiento, su presión, la permeabilidad de la roca del reservorio, etc., y puede ser que llegue a la superficie con caudales satisfactorios o no satisfactorios. Los fluidos dentro del yacimiento –petróleo, gas, agua- entran al pozo excavado, impulsados por la presión a los que están aprisionados dentro del mismo. Si la presión es suficiente, el pozo resultará “surgente”: produce sin necesidad de ayuda. Pero en la mayoría de los casos esta surgencia natural decrece y el pozo deja de producir: el pozo está ahogado.

El mecanismo de surgencia natural es el más económico, ya que la energía es aportada por el mismo yacimiento y el control de la producción se realiza en la superficie por medio del llamado “árbol de Navidad”, compuesto por una serie de válvulas que permiten abrir y cerrar el pozo a voluntad. La surgencia se regula mediante un pequeño orificio cuyo diámetro dependerá del régimen de producción que se quiera dar al pozo.

Cuando la energía natural que empuja a los fluidos deja de ser suficiente, se recurre a métodos artificiales para continuar extrayendo el petróleo. Con la extracción artificial comienza la fase más costosa de la explotación del yacimiento.

Tanto para poner a producir un pozo por *surgencia natural* así como por medios artificiales se emplean las mismas tuberías de producción “tubing”, en tramos de aproximadamente 9,45 m. de longitud, unidos por *rosca* y *cuplas*, y en distintos diámetros, desde 1,66 a 4,5 pulgadas según lo requiera el volumen de producción.

Entre los métodos de extracción artificial se cuentan los siguientes:

1. **El bombeo mecánico** que emplea varios procedimientos según sea la perforación. El más antiguo y que se aplica en pozos desde 2.400 hasta 2.500 m de profundidad, es el de la bomba de profundidad: Consiste en una bomba vertical colocada en la parte inferior de la tubería, accionada por varillas de bombeo de acero que corren dentro de la tubería movidas por un balancín ubicado en la superficie, al cual se le transmite el movimiento de vaivén por medio de la biela y una manivela, las cuales se accionan a través de una caja reductora movida por un motor. La bomba consiste en un tubo de 2 a 7,32 m de largo con un diámetro interno de 1 ½ a 3 ¾ pulgadas, dentro del cual se mueve un pistón cuyo extremo superior está unido a las varillas de bombeo. El costo promedio de este equipo asciende a US \$ 70.000 aproximadamente.
2. **Extracción con gas o Gas Lift.** Consiste en inyectar gas a presión en la tubería para aliviar la columna de petróleo y hacerlo llegar a la superficie. La inyección de gas se hace en varios sitios de la tubería a través de válvulas reguladas que abren y cierran el gas automáticamente. Este procedimiento se suele comenzar a aplicar antes de que la producción natural cese completamente.
3. **Bombeo con accionar hidráulico** Una variante también muy utilizada consiste en bombas accionadas en forma hidráulica por un líquido, generalmente petróleo, que se conoce como fluido matriz. Las bombas se bajan dentro de la tubería y se accionan desde una estación satélite. Este medio no tiene las limitaciones que tiene el medio mecánico para su utilización en pozos profundos o dirigidos.

4. **Pistón accionado a gas (plunger lift).** Es un pistón viajero que es empujado por gas propio del pozo y trae a la superficie el petróleo que se acumula entre viaje y viaje del pistón.
5. **Bomba centrífuga y motor eléctrico sumergible.** Es una bomba de varias paletas montadas axialmente en un eje vertical unido a un motor eléctrico. El conjunto se baja en el pozo con una tubería especial que lleva un cable adosado, para transmitir la energía eléctrica al motor. Permite bombear grandes volúmenes de fluidos.
6. **Bomba de cavidad progresiva.** El fluido del pozo es elevado por la acción de un elemento rotativo de geometría helicoidal (rotor) dentro de un alojamiento semi elástico de igual geometría (estator) que permanece estático. El efecto resultante de la rotación del rotor es el desplazamiento hacia arriba de los fluidos que llenan las cavidades formadas entre rotor y estator.

Otra parte importante es el extremo del pozo en la superficie, al cual se le denomina comúnmente como “cabezal” o “boca del pozo” y para el caso de pozos surgentes “árbol de Navidad”. Éste involucra la conexión de las tuberías de subsuelo con las de la superficie que se dirigen a las instalaciones de producción. El “colgador de cañerías” y el “puente de producción” son los componentes principales de la boca del pozo. Cada una de las tuberías utilizadas en el pozo, deben estar equipadas con un “colgador” que las soporte. Este colgador va enroscado en el extremo superior de la tubería y debe ser adecuado para soportar a la siguiente tubería que será de menor diámetro. Los fluidos producidos por el pozo son recibidos en la superficie en un “puente de producción”, que constituye el primer punto elemental del control de la misma. Este puente no sólo está equipado con los elementos necesarios para la producción de petróleo, junto con el gas y el agua y asociados, sino también para la captación del gas que se produce por el espacio anular entre la tubería y el revestidor.

Hasta aquí se ha tratado la extracción de petróleo en su fase de *recuperación primaria*, es decir, aquella que se efectúa en función de la energía inicial, existente en el yacimiento, acudiendo en algunos casos a métodos artificiales. Ésta fase permite obtener entre un 15% y

un 35% del petróleo. En el caso de petróleos viscosos, la extracción puede ser inferior al 10%.

Por otra parte, es común aplicar algunos medios para mejorar los valores de recuperación, como por ejemplo la inyección de gas o agua en determinados pozos denominados “inyectores”, con el objeto de desplazar volúmenes adicionales de petróleo hacia el resto de los pozos del yacimiento que conservan el carácter de “productores” y tratar con esto de extraer del yacimiento volúmenes mayores que el 10 o el 15% inicial, a eso se conoce como *recuperación secundaria*.

Existen varias razones por las cuales se realiza este tipo de recuperación:

1. **Económica:** para recuperar volúmenes adicionales de petróleo, llamados también reservas adicionales o secundarias y así, aumentar los beneficios económicos;
2. **Conservacionista:** para evitar el desperdicio de la energía natural del yacimiento;
3. **Técnica:** para reponer y mantener la presión del yacimiento.

Con respecto a la inyección de gas cabe destacar, el proyecto PIGAP (**Proyecto de Inyección de Gas a Alta Presión**), el cual inyecta gas a una presión de 9000 lb/pul² en 20 pozos inyectores del campo Santa Bárbara en Punta de Mata, estado Monagas (Venezuela), esto permitirá recuperar 32% del petróleo existente en el yacimiento (2.175 millones de barriles de petróleo) hasta el año 2035.

Además de la recuperación secundaria, se suelen aplicar otros métodos llamados de recuperación terciaria o mejorada, tales como la inyección de anhídrido carbónico (CO₂), solventes, de polímeros, o métodos térmicos tales como la inyección de vapor, o de combustión. Atendiendo a su costo elevado, esta fase se lleva a cabo cuando los precios del crudo la vuelven económicamente factible.

El petróleo, junto con el gas y el agua asociados, se lleva desde cada uno de los pozos hasta las estaciones recolectoras a través de tuberías enterradas de entre 2 y 4 pulgadas de diámetro. El material más común para estas líneas de conducción es el acero, aunque se utiliza, cada vez más, las tuberías de PVC reforzado con fibra de vidrio, resistentes a la corrosión.

Capítulo IV Metodología y Diseño

La estación recolectora recibe la producción de un determinado número de pozos del yacimiento, generalmente entre 10 y 30. Allí se cumplen funciones de separación de los diferentes fluidos, la medición diaria del volumen total producido y en los casos necesarios, de cada pozo en particular. En el caso de petróleos viscosos, también se efectúa su calentamiento para facilitar su bombeo a plantas de tratamiento.

La producción de petróleo en el mundo varía enormemente según los pozos: algunos aportan unos pocos metros cúbicos de éste fluido y otros, más de un millar por día, lo que se debe a diferentes factores como por ejemplo el volumen de hidrocarburos almacenado en el espacio poral de las rocas del reservorio, la extensión misma de las capas o estratos productivos, etc. El valor más alto de productividad se tiene en Arabia Saudita con una producción promedio de 1828,5 m³/d (11.500 b/d⁽¹⁾) por pozo. En el otro extremo se encuentra Estados Unidos con una productividad promedio por pozo de 6,4 m³/d (40 b/d).

Tabla 3: Producción promedio por pozo en diferentes países ⁽²⁾

País	b/d	m ³ /día
Estados Unidos	40	6,4
Venezuela	200	31,8
Argelia	700	111,3
Libia	1.700	270,3
Nigeria	1.750	278,3
Kuwait	4.000	636,0
Irak	7.500	1192,5
Irán	9.500	1510,5
Arabia Saudita	11.500	1828,5

Al llegar el crudo producido por los pozos, por lo general está acompañado por agua de formación, sales contenidas en el agua, sólidos en distintos tipos y tamaños y otros contaminantes peligrosos y corrosivos. Ante esta situación es necesario separar los sólidos

⁽¹⁾ Barriles diarios, unidad y nomenclatura estándar utilizada por la OPEC.

⁽²⁾ Fuente: OPEC. The Annual Statistical Bulletin (ASB) 2003

Capítulo IV Metodología y Diseño

del crudo y proceder a deshidratarlo, es decir, se elimina el agua y la sal que naturalmente contiene el petróleo en formación, o el agua que producen otras capas. Este proceso se realiza en la Planta Deshidratadora.

El hecho de acondicionar el crudo se realiza por una exigencia tanto de los transportadores, ya sea en barcos o en oleoductos, como de las refinerías, que es su destino final. Dentro de estas exigencias se establece que el petróleo no contenga un porcentaje de agua e impurezas mayor al 1% y un máximo de 100 gramos de sales por cada metro cúbico de producto.

El petróleo, una vez separado de los sedimentos, agua y gas asociados, se envía a los tanques de almacenamiento y a los oleoductos que lo transportarán hacia las refinerías o hacia los puertos de exportación.

Unidad de Producción

Para la aplicación de la teoría Holónica a un sistema de producción continua, se tiene que: un pozo representa una unidad de producción (UP).

La UP está compuesta por elementos, capaces de determinar el estado del sistema, y tomar los correctivos necesarios basándose en el mismo, según [5] los elementos que componen el sistema decisorio son:

- El observador que es responsable de la determinación del estado del proceso bajo control.
- El evaluador de alternativas quien determina, dentro de un conjunto de acciones, cuales son las que conducen a un estado deseado, asociándoles un costo con el fin de determinar la acción o secuencia de acciones óptimas.
- El regulador responsable de la generación de acciones específicas hacia el sistema controlado, de acuerdo a la selección realizada en el paso anterior.

En la capa física de la UP se tiene la representación del pozo a través de un modelo matemático simplificado y el control del flujo de salida del pozo es a través del control regulatorio.

En la capa de decisión e información se tiene a los elementos que componen el sistema decisorio representados por agentes de software, con las características nombradas anteriormente. El estado de cada uno de los dispositivos físicos y del proceso será observado por el agente de software encargado de ello, luego, con estos estados, éste agente u otro se encarga de evaluar las alternativas y de generar las acciones específicas que lleven el sistema físico a un estado admisible.

Capa Física de la UP

Modelado del Pozo

En la implementación de la capa física de la UP, se tiene un modelo simplificado del pozo, lo cual permite estimar la capacidad de producción del mismo, tomando en cuenta los estados de los dispositivos que están involucrados en el proceso.

Los dispositivos físicos del proceso para el modelado de la extracción de crudo de un pozo

son:

- 2 bombas centrífugas o eléctricas, una principal y otra secundaria. Donde cada una tiene diferentes capacidades para la extracción de crudo y 2 velocidades.
- 1 válvula para regular el flujo del crudo que sale del pozo.
- Energía.
- Tuberías.
- Inyección de Agua con flujo constante.
- Un tanque. Como modelo para la representación simplificada e ideal del reservorio⁽³⁾.

⁽³⁾ Es de hacer notar que la herramienta clásica de ingeniería de reservorios como el modelo tanque (balance de materiales que trata al reservorio como un gran tanque con propiedades promedias uniformes) no proporciona un modelado apropiado de los reservorios, sistemas hidrocarburos y/o esquemas de recuperación complejos, ya que no pueden predecir el comportamiento de los mismos. Sin embargo, para la demostración de los objetivos de éste trabajo, éste modelo es válido.

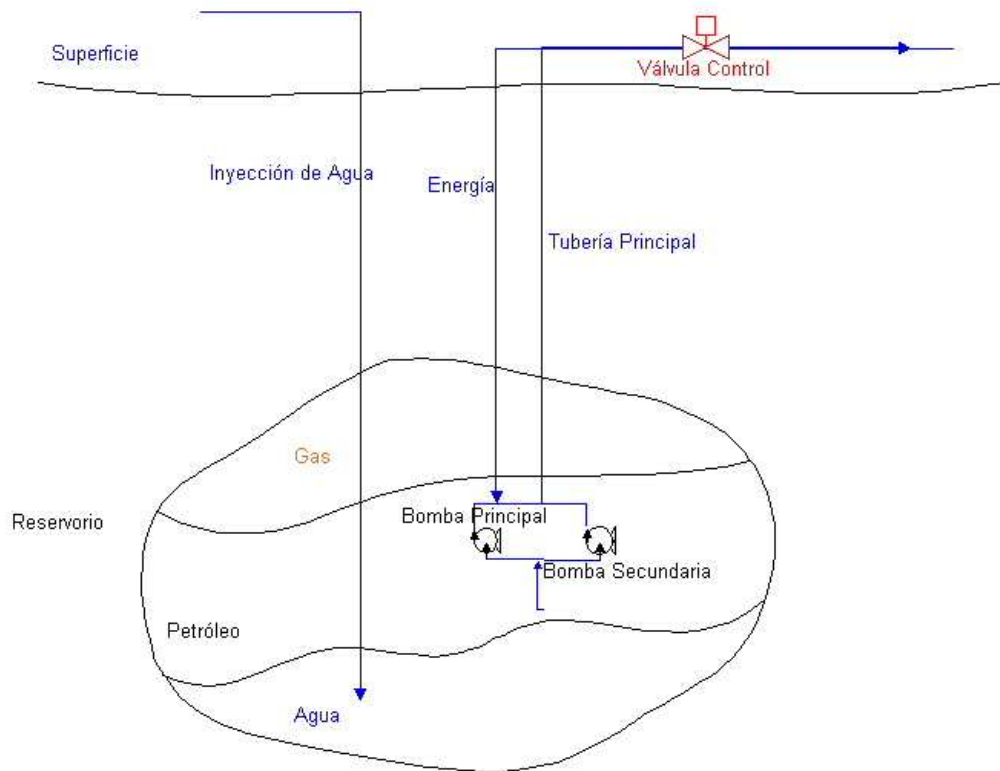


Figura 23: Proceso simplificado de extracción de crudo de un pozo de petróleo.

Para modelar el proceso se obtiene el balance de masa, que está dado por la siguiente ecuación diferencial:

$$\rho * f_1(t) - \rho * f_0(t) = \rho * A_1 * \frac{dh_1(t)}{dt} \quad \text{Ecuación (1)}$$

Donde:

ρ : densidad del líquido.

$f_0(t)$: Flujo de entrada al tanque reservorio. b/h.

$f_1(t)$: Flujo de salida del tanque reservorio.

A_1 : Área transversal del tanque reservorio (m^2).

h_1 : Altura del nivel del tanque reservorio (%)

El flujo de entrada al reservorio es agua y es constante, el flujo de salida del reservorio es petróleo, donde su densidad es: 48.62 lb/pies³ y su flujo máximo de salida está limitado por la máxima capacidad de extracción de las bombas cuando la válvula está completamente abierta, el estado actual de las bombas y el estado actual de la tubería.

Si las bombas de extracción no reciben energía (energía = 0) no puede haber extracción de flujo, igualmente si ambas bombas fallan o si las tuberías se encuentran completamente obstruidas.

Debido a que este sistema de producción es continuo las características varían en el tiempo, así que el modelo y sus parámetros (estados de los equipos y /o recursos, materia prima, energía, etc.) deben monitorearse y estimarse periódicamente, para mantener actualizados los métodos de producción, la configuración, los parámetros de control, etc.

Control de flujo de Salida del Pozo

La información del flujo de salida del pozo es almacenada localmente, en sistemas orientados al control regulatorio. Estos equipos son construidos tolerantes a fallas, con un alto rendimiento en la ejecución de lazos simples de control continuo; por lo cual, para controlar el flujo de salida del pozo se utiliza un sistema de control regulatorio como el PID.

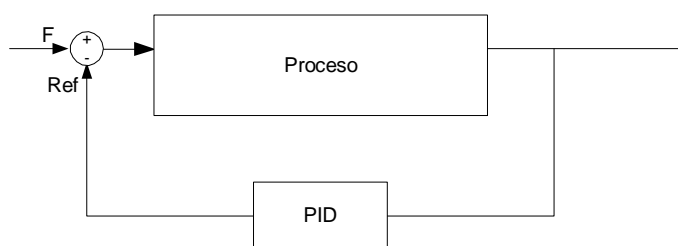


Figura 24: Control a través del PID en lazo cerrado.

Tenemos un pozo al cual hay que controlarle el flujo de salida, para ello se mide el flujo de salida del pozo y se compara con una referencia dada o setpoint, para que ambos se parezcan un controlador PID calcula el error de la diferencia y ejerciendo la acción respectiva sobre la válvula (actuador), el controlador tratará de llevar al mínimo ese error. Este tipo de

control puede controlar solo una parte del sistema, dado que su visión es local, pero en cambio si se tiene algún tipo de control supervisorio que contenga una vista global de todo el proceso, se podrán tomar otro tipo de decisiones en caso de que lo que suceda escape del alcance del PID clásico, ésto es lo que se busca en la siguiente capa.

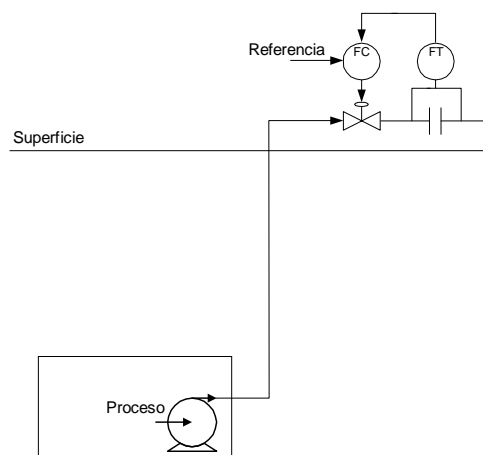


Figura 25: Control del Flujo de Salida del Pozo

Capa de Decisión e Información de la UP

En esta capa se realizan las actividades de observación - determinación del estado del proceso, evaluación de alternativas y selección de alternativas mediante el conocimiento del sistema y de los valores actuales del proceso, los cuales se obtienen mediante los sensores instalados en el proceso. Los datos representativos del proceso, almacenados en su imagen, permiten determinar el estado presente e inferir resultados. Ésta tarea debe realizarse de manera cíclica.

El sistema supervisorio, para éste caso, representado en un sistema multiagentes, es responsable de determinar las acciones a ser ejecutadas, con el fin de conducir al sistema controlado hacia un estado predeterminado.

El sistema decisorio funciona en tres etapas:

1. Evaluación del estado del proceso a través del estudio de las variables del proceso o de los eventos que ocurren en el mismo,

2. Evaluación de alternativas posibles a partir del estado actual,
3. Fijación de las secuencias de acciones que conducen al sistema hacia el objetivo deseado.

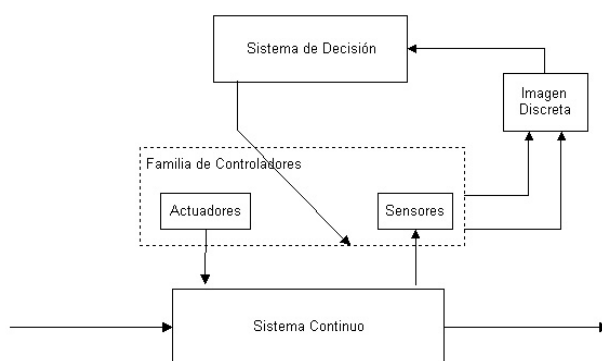


Figura 26: Arquitectura de Supervisión.

En la figura anterior, podemos observar que el sistema de decisión, toma, a través de los sensores, los datos del proceso y los almacena en una imagen discreta, esto, suministrará la información necesaria que le permitirá al sistema de decisión determinar el estado actual del proceso, inferir resultados y actuar sobre el proceso en base a ello. La conversión se hace a través de los Sistemas a Eventos Discretos (SED), y se modela a través de las redes de Petri. Un ejemplo de esto, podría ser lo siguiente: un sistema de eventos discretos transformará todo el conjunto de temperaturas que se encuentren dentro del rango de 200°C y 220°C (parte continua), en un estado llamado: *normal* (imagen discreta), tal que, si alguna lectura de una temperatura fuera por ejemplo de 205.3764°C , el sistema de eventos discretos tendría la imagen de: *normal*, esto hace mucho más fácil saber el estado actual de la variable y tomar decisiones basándose en ello.

Tomando en cuenta lo anterior, se explicaran los estados discretos de cada uno de los dispositivos involucrados en el proceso, lo cual nos permitirá obtener la imagen actualizada del estado del proceso, para evaluar su estado actual y tomar las acciones que sean necesarias.

Estados discretos para las bombas

En la simulación se tienen dos bombas, una bomba principal y otra secundaria o auxiliar, de forma tal de reducir la posibilidad de parar la producción en caso de que alguna bomba falle. Cada una tiene 2 velocidades diferentes y 5 posibles estados discretos:

- Normal
- Degradada
- Disponible
- Mantenimiento
- Obstruida/ Falla

Cuando la bomba esta en estado *normal* su capacidad de extracción es el 100% del flujo de extracción actual, ese flujo de extracción dependerá de la configuración de la velocidad que se tenga para ese momento. En éste estado, su funcionamiento es óptimo. Cuando el estado de la bomba es *degradada*, su desempeño disminuye, esto podría deberse a obstrucciones parciales o fallas de naturaleza estocástica, y aunque está funcionando no extrae el 100% del flujo que debería, sino una cantidad inferior (entre el 3% y el 98% de su capacidad actual). En el estado *disponible*, la bomba esta al 100% de su capacidad, sin embargo no está siendo utilizada, pero está disponible para que cuando sea requerida se use, es decir, se cambie a estado *normal*. En *mantenimiento*, la bomba esta siendo reparada después de una falla (mantenimiento correctivo), o está siendo mantenida de acuerdo a alguna agenda (mantenimiento preventivo), de forma tal que se le alargue su vida útil; en éste estado no puede hacerse uso de la misma. Si la bomba se encuentra en el estado *Obstruida/ falla* probablemente es porque ocurrió alguna falla u obstrucción total de naturaleza estocástica o porque debido a la falta de mantenimiento preventivo, se dañó; en este estado la bomba no está en uso.

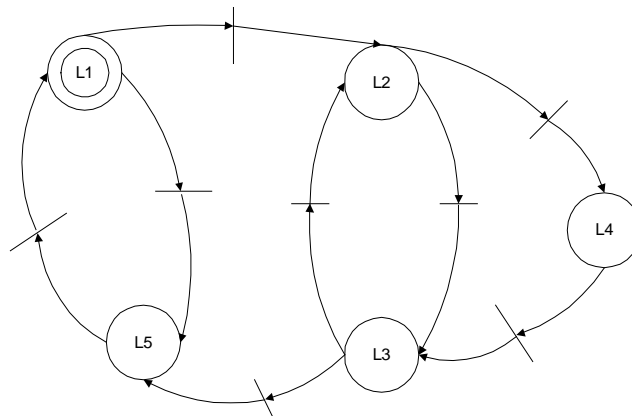


Figura 27: Red de Petri para los estados de la bomba

De los estados anteriores, podemos decir que solo 3 estados pueden ser controlados por el sistema decisorio: **normal, mantenimiento, disponible**, según [5] “*Un evento se considera controlable si el mismo puede ser deshabilitado por un supervisor, con el fin de que no aparezca en el sistema, mientras que los elementos no controlables (incontrolables), aparecen espontáneamente en el sistema discreto, y la única forma de evitar que aparezcan, es mantener el sistema en estados donde los mismos no pueden presentarse*”, es decir que el sistema decisorio solo habilitará o deshabilitará los estados que se pueden controlar y mantendrá la supervisión de forma tal, que los estados no controlables: falla y degradada no puedan presentarse.

Tabla 4: Descripción de la Red de Petri para los estados discretos de la bomba.

Estado	Descripción
L1	Normal
L2	Degradada
L3	Mantenimiento
L4	Falla
L5	Disponible

No ha sido objetivo de este trabajo, simular las fallas o mantenimientos de los equipos, por lo cual se ha dispuesto de una pantalla independiente que puede manipularse para simular este tipo de eventos (ver apéndice ¿??).

Anteriormente se dijo, que en la simulación se tienen dos bombas (la principal y la secundaria o auxiliar) y que cada una presenta dos velocidades distintas de extracción, la decisión del uso de cierta bomba y la velocidad de su funcionamiento, dependerá de la visión global que tenga el agente encargado de escoger a la misma⁽⁴⁾.

Estados discretos del tanque reservorio

Al igual que las bombas, el tanque que representa al reservorio, también tiene estados discretos, dependiendo de su porcentaje de altura, representados a través de una Red de Petri.

Los posibles estados del reservorio son:

- Nivel Normal
- Baja Producción
- Baja Baja Producción
- Sin Petróleo

Se asume que el estado del tanque reservorio es *normal*, en la primera etapa de extracción del yacimiento o recuperación primaria, ya sea que presente surgencia natural o si no la tiene, por alguno de los métodos de extracción artificial, descritos anteriormente, esto representa, para el caso de estudio un 20% de su capacidad de producción, es decir, cuando la capacidad del yacimiento se encuentra entre 100% y 80%. Al terminar la etapa anterior se pasa a la recuperación secundaria, la etapa más costosa ya que se depende totalmente de un método artificial, es en esta etapa cuando el estado del tanque reservorio pasa a *baja producción*, esta representa la segunda etapa y abarca un 30% de la producción total, es decir, para el caso de estudio, entre el 80% y el 50% de su capacidad inicial. La tercera etapa de extracción o recuperación terciaria se da, si y solo si, el precio de venta de petróleo en el

La explicación detallada se da en la Figura 33.

mercado es lo suficientemente alto como para recuperar la inversión y en algunos casos puede abarcar hasta un 30% más de recuperación, lo que significa, para el caso de estudio, entre el 50% y el 30% de la capacidad inicial, acá el estado del tanque reservorio se encuentra en *Baja Baja Producción*. Al finalizar la etapa de baja baja producción se asume que el estado del tanque reservorio es *sin petróleo*.

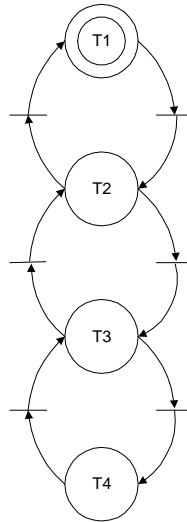


Figura 28: Red de Petri para los estados del tanque reservorio

Tabla 5: Descripción de la Red de Petri para los estados discretos del tanque reservorio

Estados	Descripción
T1	Nivel Normal
T2	Bajo nivel de producción
T3	Bajo Bajo nivel de producción
T4	Sin petróleo

Ninguno de los estados anteriores es controlable, sin embargo, el sistema decisorio debe observarlos ya que la misión principal del holón o unidad de producción es la extracción

de petróleo y cuando una unidad de producción no tiene petróleo para extraer, el ciclo de vida ha llegado a su final.

Para que el sistema decisorio, pueda saber cual es el estado actual del reservorio, establece una comunicación con el sensor en el pozo, toma el valor de la variable continua que representa a la capacidad y una vez que se tiene, se evalúan los diferentes rangos de los estados para saber a cual pertenece. Si la variable continua se encuentra entre 100 y 80% pertenece al estado *normal*, si se encuentra entre 79,99 y 50%, pertenece al estado *bajo nivel de producción*, si se encuentra entre 49.99 y 20% pertenece al estado *bajo bajo nivel de producción*, si no es alguna de las anteriores el estado del pozo será: *sin petróleo*.

Estados discretos de la tubería

El estado de la tubería también puede ser representada en forma discreta a través de una red de Petri como se muestra en la siguiente figura:

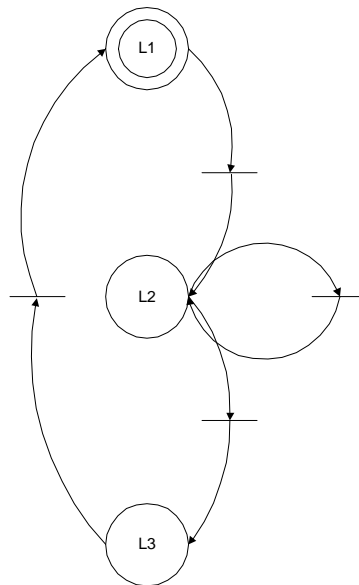


Figura 29: Red de Petri para los estados de la tubería.

Tabla 6: Descripción de la Red de Petri para los estados discretos de la tubería

Estado	Descripción
L1	Normal
L2	Degradada
L3	Obstruida

Estos estados pueden ser deducidos por el sistema decisorio a través de la evaluación de los estados de otras variables, como son el estado de las bombas, el estado de la válvula, el setpoint aplicado en ese momento, etc. Los estados de la tubería son todos no controlables, es decir, aparecerán espontáneamente sin que el sistema decisorio pueda hacer algo para cambiarlos.

Estados discretos de la válvula

Al igual que los equipos anteriores, los estados de la válvula también pueden representarse a través de un sistema a eventos discretos. La válvula sólo presentará dos estados: abierta o cerrada y **ambos estados son controlables a nivel del sistema supervisorio**. Ahora bien, la apertura de la válvula es administrada por el controlador PID.

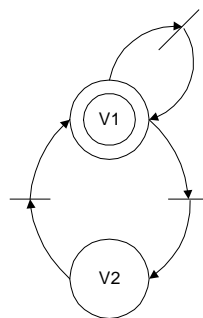


Figura 30: Red de Petri para los estados de la válvula.

Tabla 7: Descripción de la Red de Petri para los estados discretos de la válvula.

Estado	Descripción
V1	Abierta
V2	Cerrada

Estados discretos de la Energía

Los estados de la energía permiten establecer cuando el pozo está funcionando o no, ya que si no hay energía (E2) las bombas no pueden extraer crudo y la válvula se cierra (FC) y por lo tanto se presenta un estado crítico dentro del pozo. La energía presenta dos estados: E1, donde si hay energía y E2 cuando no hay energía, ambos estados son no controlables para el supervisor.

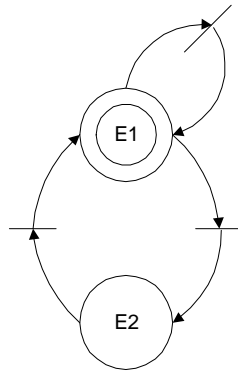


Figura 31: Red de Petri para los estados de la Energía

Tabla 8: Descripción de la Red de Petri para los estados discretos de la Energía

Estado	Descripción
E1	Si
E2	No

La combinación de los diferentes estados de los dispositivos permite obtener el estado actual del proceso. Esto a su vez, permite llevar a cabo, la supervisión y el control discreto del proceso.

Control discreto de las bombas

Con el objetivo de mantener la producción actual lo mas cerca posible de la producción esperada (setpoint) se utiliza un PID como controlador local, sin embargo, cuando se produce una falla, fuera del alcance de éste, falla en la tubería, falla en la energía, falla en las bombas, etc., el PID ya no será suficiente para mantener controlada la producción. Como segunda estrategia de control, pero ya en una capa superior, como es la de Decisión e Información, podemos llevar a cabo el control discreto en las bombas, de forma tal que si el estado de la bomba actual no satisface la producción esperada, se pueda utilizar la otra bomba y si a su vez, el estado de la segunda bomba, tampoco la satisface, se pueda llevar a cabo una negociación para lograr así el objetivo.

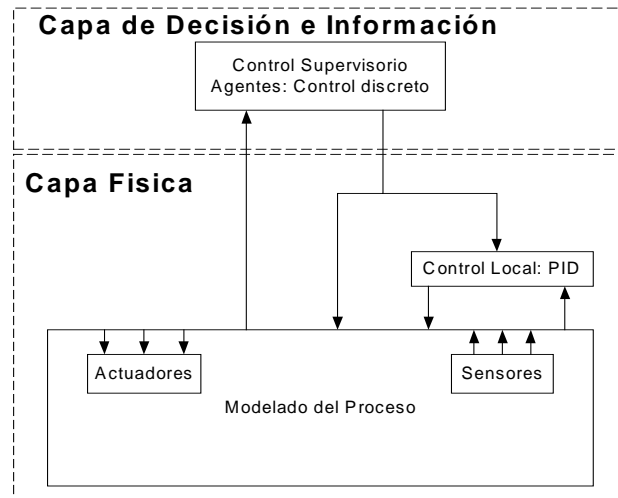


Figura 32: Capa donde se lleva a cabo el control discreto de las bombas

El control discreto, es llevado a cabo gracias al agente encargado de supervisar, evaluar y llevar a cabo las decisiones que corrigen la falla presentada.

Inicialmente se deben obtener los estados actuales de cada una de las bombas y la bomba usada actualmente, dependiendo de lo anterior se evalúan las diferentes alternativas y basándose en las mismas, se llevan a cabo los correctivos necesarios para poder mantener la producción actual lo mas cerca posible de la esperada. La siguiente figura nos muestra, a través de un diagrama de flujo, las diferentes alternativas según el estado de las bombas y las acciones a seguir:

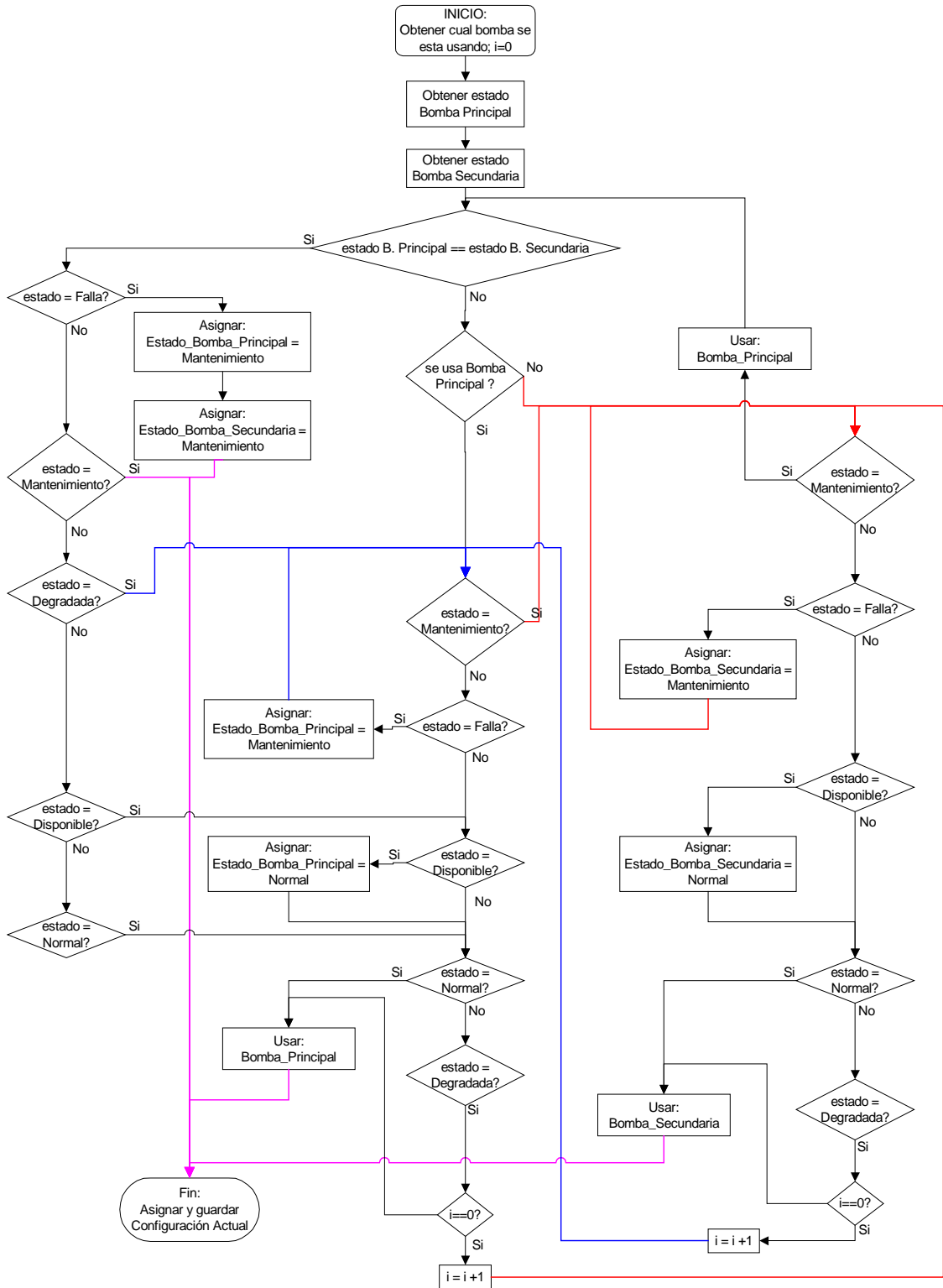


Figura 33: Diagrama de flujo para el control discreto de las bombas.

Si como consecuencia el estado resultante de alguna o de ambas bombas fuera de degradación y/o mantenimiento y no se pudiera conseguir la producción esperada, el agente encargado de controlar las bombas debe percatarse de esto y tomar la acción necesaria, pero ya en un nivel superior (enviar un alerta, negociar la producción faltante, etc.).

Estados discretos de la producción del proceso

Hay que tener en cuenta que el proceso evoluciona y/o cambia de estado en base a la composición de los diferentes estados discretos que presenta en sus dispositivos.

De la composición de los diferentes estados de los dispositivos podemos decir que los estados de la producción serán los siguientes:

- Producción Alta
- Producción Media
- Producción crítica

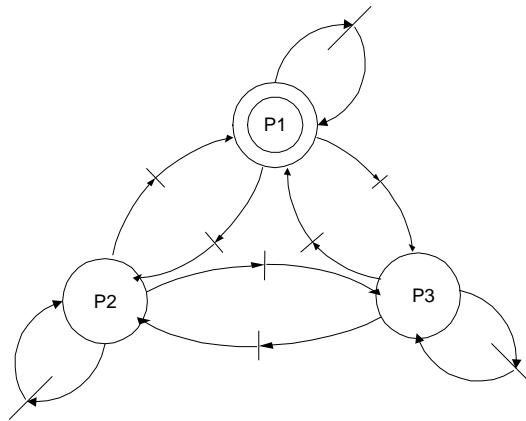


Figura 34: Red de petri para los estados de la producción

Tabla 9: Descripción de la Red de Petri para los estados discretos de la producción

Estado	Descripción
P1	Producción Alta
P2	Producción Media
P3	Producción Baja

P1 sucede sí:

- El estado de la energía es SI,
- El estado de la tubería es Normal o degradada y la degradación es menor al 20%,
- El estado de por lo menos 1 bomba es normal o degradada (y la degradación es menor al 20%) y por lo tanto pueden extraer la producción esperada y más,
- El estado del reservorio es Normal (entre 80% y 100%),
- La válvula esta funcionando dentro del rango de apertura para la que fue dimensionada y por lo tanto se está obteniendo la producción esperada (setpoint).

P2 sucede cuando:

- 1 de las bombas esta en estado de falla, mantenimiento o degradada y la otra se encuentra degradada o normal y se puede seguir obteniendo la producción esperada,
- El estado de la energía es SI

Capítulo IV Metodología y Diseño

- La tubería está degradada, pero al cambiar las velocidades de las bombas se puede seguir obteniendo la producción esperada,
- La válvula está funcionando fuera del rango para la que fue dimensionada, pero se está obteniendo la producción esperada,
- El estado del reservorio es de baja producción (entre 80% y 50%)

El último estado se da cuando:

- Ninguna de las dos bombas se puede usar (Mantenimiento o falla), o el porcentaje de degradación no permite que salga la producción esperada,
- La energía en el pozo es NO,
- El estado de la tubería sea degradada y el porcentaje de degradación no permita que salga la producción esperada (setpoint), o que el estado sea obstruida,
- El estado del reservorio sea baja baja o sin petróleo (menor al 50%),
- La válvula esté cerrada o abierta al 100%,
- No exista forma alguna de producir lo que se espera.

Comportamiento de una Unidad de Producción

La UP debe comportarse en concordancia a los eventos que puedan presentarse en el proceso, es decir, la conducta de la UP se desarrollará en base a la composición de los diferentes eventos discretos presentados en cada uno de los componentes del proceso y del acoplamiento que tengan éstos, con el sistema de control discreto (supervisor), el híbrido (discreto y continuo) y la evolución del proceso en sí.

La conducta discreta global de una UP puede ser modelada por una red de Petri cuya dinámica cambia como resultado de los eventos ocurridos. En la siguiente figura se puede observar la evolución de una UP genérica. Los nodos de la transición reciben y envían mensajes asociados a los eventos. Dentro de los eventos normales, el evento de arranque del

proceso se activa para empezar el proceso de producción, si todas las condiciones necesarias para ejecutarlo han sido satisfechas.

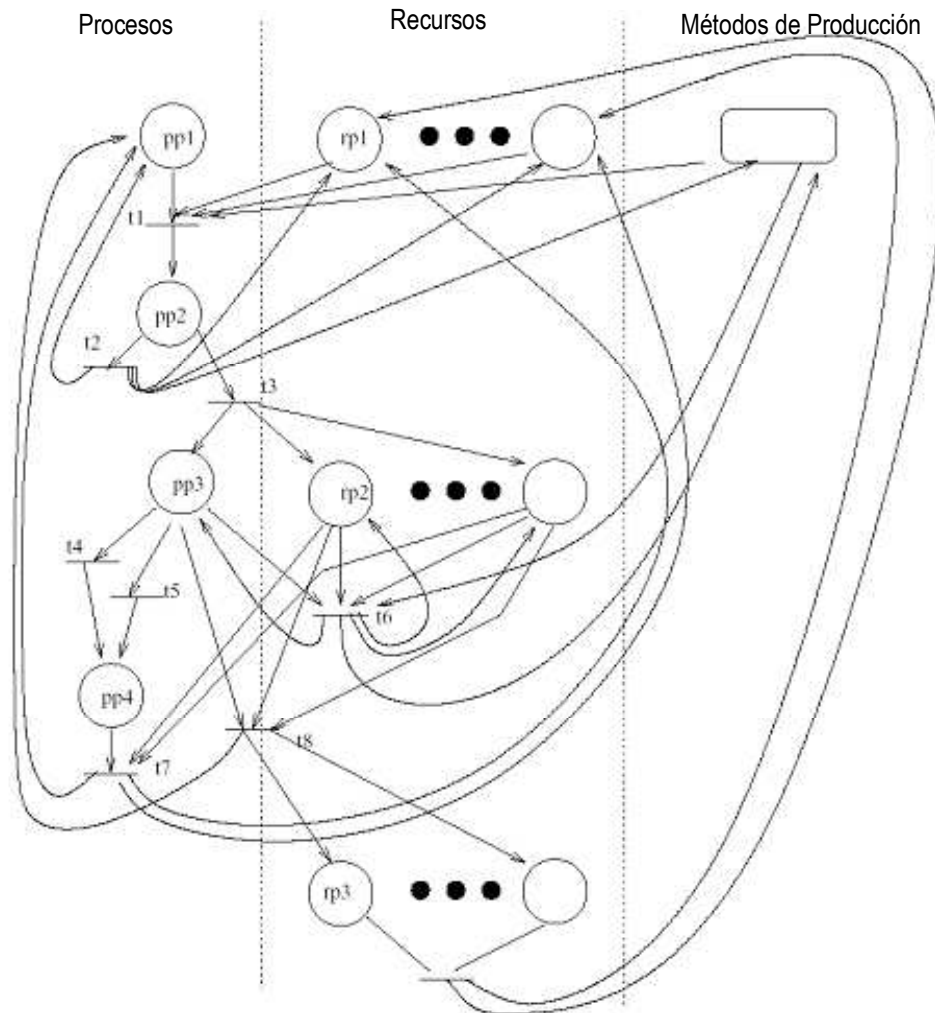


Figura 35: Comportamiento de una unidad de producción genérica [3].

Tabla 10: Descripción de la red de petri para los nodos del comportamiento de una UP

Red de Petri	Nodo	Descripción
Proceso	pp1	Sin proceso
	pp2	Evaluación
	pp3	Operación
	pp4	Apagando
Recursos	rp1	Disponible
	rp2	En Uso
	rp3	En Mantenimiento
Métodos de Producción		Existen

Agentes

Para lograr que la unidad de producción evolucione según los cambios dinámicos en los estados del proceso hay que lograr que los cambios en éstos, sean detectados, evaluar la influencia sobre el proceso de los cambios de estado y llevar a cabo las acciones que sean necesarias. Para ello, se implementan 4 agentes tal como sugiere [3], cada uno de ellos llevará a cabo tareas específicas, en forma coordinada, tal que les permitirá lograr los objetivos de la UP.

Estos agentes son:

Agente Supervisor, Agente Negociador, Agente de Recursos y Agente de control

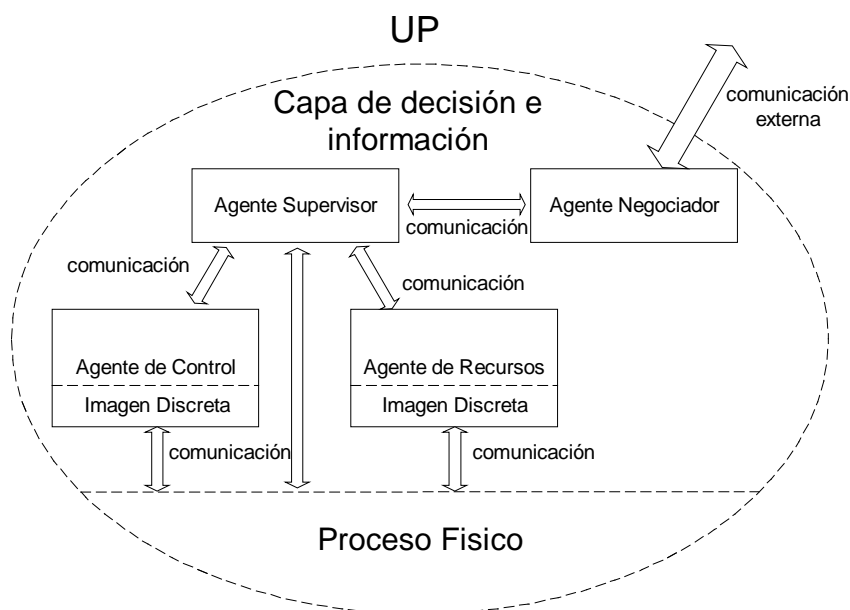


Figura 36: Agentes que componen la UP y sus comunicaciones

Los agentes de control y de recursos, son observadores en el sistema, así que se comunican periódicamente con los sensores del proceso a fin de obtener la imagen discreta del mismo y por lo tanto el estado actual en el que se encuentra, basándose en la evaluación de éste estado, llevarán a cabo alguna acción. Éstos agentes pueden comunicarse con el agente supervisor de la UP para comunicar, solicitar o responder alguna petición.

Por otra parte, el agente supervisor de la UP, también puede comunicarse con el proceso periódicamente, a fin de evaluar la evolución de los objetivos o metas principales y generales de la UP, y comunicarse con los agentes de recurso, control y negociación para solicitar o responder alguna petición. Por último el agente negociador, se encarga de negociar alguna solicitud hecha por el agente supervisor de la UP o por otros agentes negociadores, externos a la UP.

Cada agente, tiene asignadas metas particulares en concordancia con la meta global de la UP, para alcanzarlas, es necesario que tengan los comportamientos adecuados y que sean capaces de comunicarse con los otros agentes para colaborar entre sí. Esto significa que dentro de la UP, tendremos a un sistema multiagentes (MAS, por sus siglas en inglés) que

deberá actuar dinámicamente de forma tal, que hagan que la UP tenga las características holónicas.

Agente de Control

Este agente es el encargado de verificar que los parámetros del proceso se encuentren controlados, para ello supervisa los dispositivos de control regulatorio y les asigna los parámetros adecuados de la ley de control para el proceso. Por lo tanto sus objetivos específicos son:

- Asignar, verificar y reestablecer la producción esperada al nivel de planta.
- Asignar los parámetros de funcionamiento del PID.
- Mantener una comunicación con el agente supervisor, para informar, solicitar o llevar a cabo acciones sobre la producción esperada o PID.

Agente de Recursos

Este agente se encarga de asignar la configuración de los recursos según la producción asignada a la UP, supervisa el correcto funcionamiento de los recursos físicos de la UP e intenta reestablecerlos en caso de alguna falla, para ello lleva a cabo las siguientes tareas:

- Observar, controlar y fijar la configuración de cada bomba.
- Observar el estado de los diferentes recursos controlables o no (tubería, pozo, energía, válvula), guardar, actualizar y analizar el estado de cada uno e informar sobre algún estado no deseado.
- Calcular e informar el estado global de los recursos y configuración actual.
- Mantener una comunicación con el agente supervisor para informar, solicitar o llevar a cabo acciones sobre los recursos.

Agente Supervisor de UP

Básicamente, este agente es el encargado de supervisar al resto de los agentes y comprobar que las metas globales de la UP se estén cumpliendo. Para ello:

- Solicita la información sobre el estado actual de los recursos, en forma general o particular, al agente de recursos.
- Solicita información sobre el estado del control, al agente de control.
- Solicita cambios en las configuraciones de los recursos y/o la producción esperada, en base a algún cambio interno o externo de la UP (nueva negociación en la producción, falla en algún recurso, etc.).
- Calcula, basándose en las configuraciones posibles de los recursos y el estado actual de los mismos, la producción disponible de la UP.
- Calcula el estado actual de la UP, sobre la base de la información solicitada a los agentes de control y de recursos.
- Observa los cambios que ocurren en la producción de la UP e implementa los correctivos necesarios, para ello podría solicitar a los agentes respectivos cambios en las configuraciones sobre los recursos, producción esperada y/o negociar con otras UP.

Agente Negociador

De los agentes de una UP, éste agente es el único que puede comunicarse con agentes externos a la UP para negociar y tomar la decisión de aceptar o no, basándose en la información recibida del agente supervisor. Por lo tanto sus tareas específicas son:

- Iniciar y /o responder negociaciones con agentes externos a la UP para balancear y/o reestablecer la producción según la meta, de la propia unidad o del conjunto de UPs. Cabe destacar que éste agente solo inicia una negociación cuando el agente supervisor de pozo le hace la solicitud.

- Informar al agente supervisor de zona, las decisiones que se produzcan en alguna negociación y los cambios en la producción de la propia UP, como consecuencia de la misma.

Agente Supervisor de Zona

Este agente es el encargado de supervisar todas las UPs de la zona y de resguardar que la meta de producción total de la zona se esté cumpliendo.

Una zona contiene a varias UP dentro de ella y la producción total de la misma, será la suma de la producción de cada UP perteneciente a la zona.

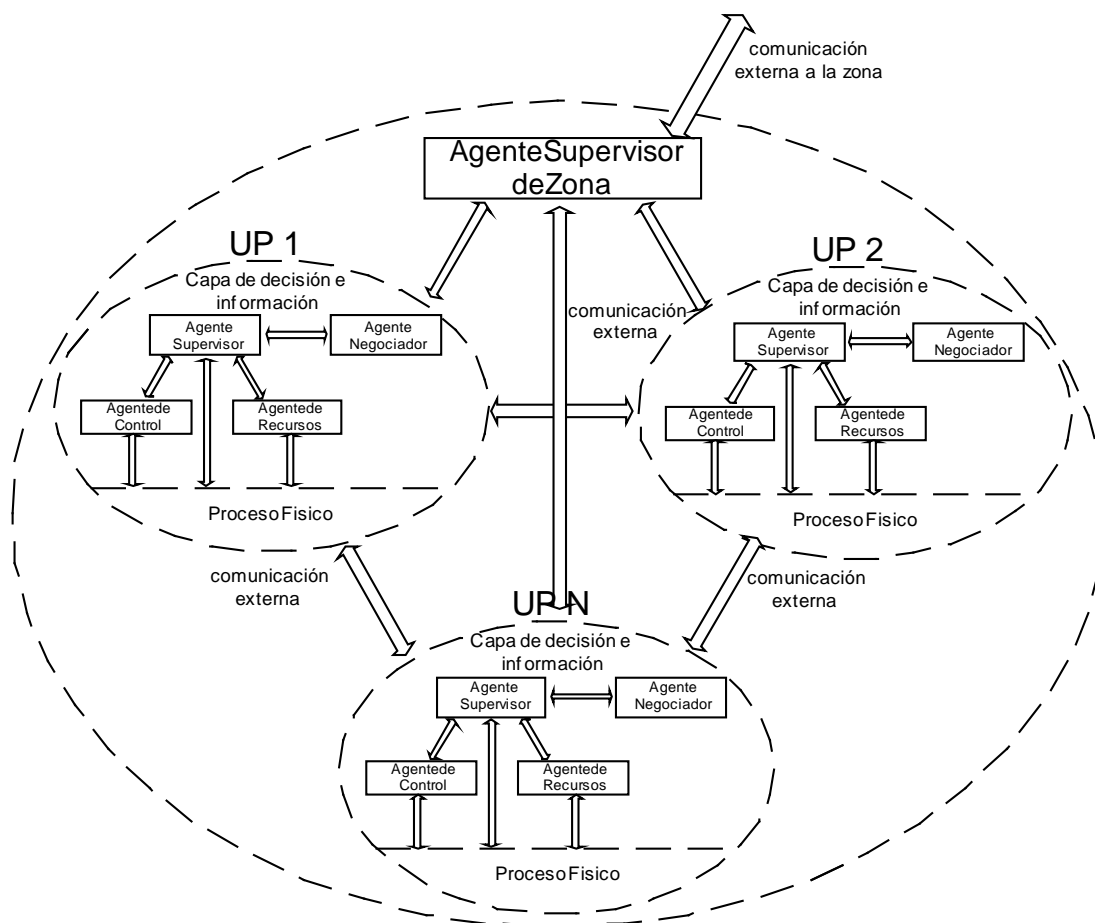


Figura 37: Interacción entre el agente supervisor de zona y las UPs pertenecientes a la zona.

Capítulo IV Metodología y Diseño

Para éste diseño, un usuario introduce alguna cantidad de producción diaria para la zona, y es el agente supervisor de zona quien se encarga de negociar con cada UP la porción que puede producir cada una, cuando esto suceda el agente calculará la producción total de la zona e informará al usuario si la cantidad solicitada inicialmente se puede o no llevar a cabo.

Dentro de las tareas particulares para éste agente se tienen:

- Interactuar con el usuario para la asignación de la producción total de la zona.
- Negociar con las diferentes UP de su zona, para solicitar algún tipo de cambio en la producción dada por un usuario.
- Mantener actualizada la producción total de la zona, solicitando a cada UP su producción y/o reflejando algún cambio en la producción en alguna UP.

Comunicación inter agentes en la UP

Para mantener la información sobre la producción de la zona y de las UPs correcta y actualizada, los agentes mencionados anteriormente deben mantener una comunicación constante a través del intercambio de mensajes.

FIPA establece el estándar llamado ACL, donde, el mensaje contiene ciertos parámetros o campos que deben asignarse para que tenga sentido para el agente que lo recibe. En el diseño para este trabajo los campos obligatorios a asignar por los agentes son: Sender, Receiver, Performative, Content, Language, Protocol, Ontology, Conversation-id, y por último Reply- with⁽⁵⁾.

Los protocolos escogidos, la ontología creada y el intercambio de mensajes entre los agentes, deben ser tales, que expresen espontáneamente la intención sobre la acción o información a solicitar.

Protocolos

La organización FIPA ha desarrollado algunos protocolos estándares de forma tal, que en su performativa, lleva implícita la intención de la solicitud o la respuesta, así que

⁽⁵⁾ Ver capítulo III, tabla 1.

diferentes agentes de software, desarrollados por diferentes programadores, podrían entenderse si adoptan el estándar. Por otra parte los estándares de FIPA son los más usados por los diferentes programadores de agentes de software en todo el mundo y ha sido una de las principales razones para escogerlos.

Cuando un agente, de los nombrados anteriormente, necesita solicitar, al agente que va a recibir el mensaje, que lleve a cabo alguna acción (p.e.: Solicitar alguna información en particular o pedirle que haga alguna tarea) utilizará el protocolo de interacción estándar llamado: FIPA Request interaction protocol, el agente que reciba el mensaje ACL responderá a la solicitud hecha en el campo del contenido del mensaje.⁽⁶⁾

Si algún agente desea proponer que él mismo va a llevar a cabo alguna acción se comunicará con el otro agente a través del protocolo de interacción estándar llamado: FIPA Propose interaction protocol, para solicitarle que hará lo que se encuentra en el campo del contenido del mensaje ACL enviado⁽⁷⁾.

Para negociar (solicitar a los agentes que participen en la negociación que lleven a cabo alguna labor) el agente iniciador podrá hacerlo a través de dos protocolos estándares: FIPA Contract Net Interaction Protocol⁽⁸⁾ y FIPA Iterated Contract Net Interaction Protocol⁽⁹⁾. El primero lo usarán para llevar a cabo negociaciones internas en la UP, el segundo será para las negociaciones externas a la UP (negociación con otras UPs o con el supervisor general de zona). El objeto de la negociación estará especificado en el campo del contenido del mensaje ACL.

Ontología.

La ontología tiene que ver con la necesidad de definir un vocabulario y una semántica propia para el contenido de los mensajes que se intercambian entre los agentes. Solo así, definiendo una ontología, el mensaje que recibe un agente tiene significado para él.

⁽⁶⁾ Ver capítulo III, figura 7.

⁽⁷⁾ Ver capítulo III, figura 16.

⁽⁸⁾ Ver capítulo III, figura 10.

⁽⁹⁾ Ver capítulo III, figura 11.

JADE proporciona tres formas de llevar a cabo la comunicación entre agentes:

La primera y más básica, consiste en usar cadenas de caracteres para representar el contenido de los mensajes. Esto es conveniente cuando el contenido del mensaje es de datos puntuales, pero no en el caso de conceptos abstractos, objetos o estructura de datos. En tal caso, las cadenas necesitarían ser analizadas para acceder a todas las partes del objeto o de la estructura de datos.

La segunda manera aprovecha la tecnología de Java para transmitir objetos Java directamente serializados como el contenido de los mensajes. Esto es un método conveniente para una aplicación local donde todos los agentes están implementados en Java. Un inconveniente, es que estos mensajes no pueden ser leídos por los humanos.

La tercera manera involucra la definición de los objetos a ser transferidos como extensión de una clases predefinida para que Jade pueda codificar y decodificar los mensajes en un formato estándar de FIPA. Esto le permite a los agentes jade ínter operar con otros sistemas de agentes que no sean JADE, lo cual le da una gran ventaja al diseño propuesto, por lo tanto, ésta será la manera de implementar éste diseño.

Los agentes tienen la capacidad de solicitar a otros agentes que lleven a cabo alguna acción, para ello utilizan algún protocolo estándar y una ontología predefinida por el programador. Las acciones a solicitar deben ser consistentes con las tareas que el propio agente realiza, por ejemplo, si se desea asignar un nuevo valor en la producción, la solicitud debe hacerse al encargado de manejar esto (el agente de control) y no a algún otro agente.

Así que para cada agente se tendrán mensajes estructurados, especificados en una clase en java que define la ontología para todos los agentes, e involucrará los conceptos y acciones que deben incluirse y definirse en la ontología.

Ontología para el Agente de Control

Cuando se le solicite llevar a cabo la acción de Asignar producción al nivel de planta el mensaje debe estar estructurado de la siguiente forma:

Capítulo IV Metodología y Diseño

[(acción) AsignarSetPoint, (concepto: flotante) valor] ambos son obligatorios dentro del contenido del mensaje,

Cuando éste agente informe sobre fallas en la producción, deberá enviar un mensaje con la siguiente estructura:

[(acción) Informar; (concepto) causa; (concepto) Falla en Producción; (concepto) Producción restablecida.]. De los campos del mensaje anterior, informar, causa y falla en la producción son obligatorios y el concepto producción restablecida es opcional, ya que solo se utilizará cuando la falla ya no exista.

Ontología para el Agente de Recursos

Cuando éste agente informe sobre algún estado no deseado en alguno de los diferentes recursos, deberá enviar un mensaje con la siguiente estructura:

[(acción) Informar; (concepto) causa ; (concepto) Falla en Recursos; (concepto) Recursos restablecidos]. De los campos del mensaje anterior, informar, causa y falla en recursos, son obligatorios y el concepto recursos restablecidos es opcional, ya que solo se utilizará cuando la falla ya no exista.

Cuando éste agente informe sobre el estado global de los recursos y la configuración actual, deberá enviar un mensaje con la siguiente estructura:

[(concepto)Bombas: [[(concepto) Bomba Principal: (entero) estado de la bomba, (flotante) flujo Máximo; (flotante) flujo Mínimo; (flotante) flujo Usado]; [(concepto)Bomba Secundaria: (entero) estado de la bomba, (flotante) flujo Máximo; (flotante) flujo Mínimo, (flotante) flujo Usado]; (concepto: entero) bomba Usada;]

(concepto) Tubería: [(entero) estado de la tubería, (flotante) degradación];

(concepto) Energía: [(entero)estado de la energía];

(concepto) Pozo: [(entero) estado del pozo]].

Cuando a éste agente se le solicite llevar a cabo alguna acción sobre los recursos, ésta acción sólo debe incluir a aquellos recursos cuyos estados pueden ser controlados, por lo tanto, los únicos recursos sobre los que se puede ejercer alguna acción son las bombas y las

únicas acciones que pueden ser ejercidas por un agente externo al agente de recurso es cambiar de bomba o cambiar de velocidad en la bomba, pero pasar una bomba de falla a mantenimiento, o de normal a disponible o a mantenimiento, solo puede ser ejercida autónomamente por el agente de recursos, sin que algún otro agente tenga ningún tipo de intervención sobre esto. Así que el mensaje deberá referirse solamente a cambiar la bomba o velocidad y estar estructurado de la siguiente manera:

[(acción) Tipo de acción, (concepto) valor]. De los campos del mensaje anterior, ambos son obligatorios

Ontología para el Agente Supervisor de UP

Cuando éste agente desee solicitar información sobre el estado actual de los recursos, en forma general, colocando todos los campos de todos los recursos, o de forma particular, colocando solo los campos de los recursos que le interesan, deberá escribir un mensaje estructura de la siguiente forma:

[(acción) Consultar Recursos; (concepto) (campos a consultar⁽¹⁰⁾)].

Cuando éste agente desee solicita información sobre el estado de la producción al agente de control, deberá incluir un mensaje con la siguiente estructura:

[(acción) Información, (concepto: flotante) valor del setpoint] ambos son obligatorios dentro del contenido del mensaje,

Cuando éste agente desee solicita cambios en las configuraciones de los recursos y/o la producción esperada, el mensaje deberá tener la misma estructura descrita anteriormente para los agentes de recursos y/o de control, respectivamente.

Cuando éste agente desee solicitarle al agente negociador que lleve la acción de negociar, deberá enviarle un mensaje estructurado de la siguiente forma:

[(acción) Negociar; (concepto) identificación de la petición; (concepto: flotante) cantidad a negociar]. Todos los campos anteriores son obligatorios.

⁽¹⁰⁾ Los campos a consultar estarán estructurados de la misma forma que se definen en la *ontología para el agente de recursos*.

Ontología para el Agente Negociador

Cuando este agente desee iniciar y /o responder a alguna negociación con agentes externos a la UP deberá enviar un mensaje con la siguiente estructura:

[(acción) Petición; (concepto) dirección en la Producción: [(concepto) Aumentar / Disminuir]; (concepto: flotante) cantidad; (concepto: flotante) configuración Bomba principal Mínima; (concepto: flotante) configuración Secundaria Mínima; (concepto: flotante) configuración Principal Máxima; (concepto: flotante) configuración Secundaria Máxima]. Todos los campos anteriores son opcionales.

Cuando éste agente desee informar al agente supervisor de zona, las decisiones que se produzcan en alguna negociación deberá estructurar el mensaje de la misma forma que utiliza el agente supervisor, para solicitarle que negocie.

Mensajes Inter Agentes.

Al unir el protocolo que se debe adoptar para comunicar intuitivamente la intención de la solicitud y la estructura del mensaje que se debe enviar, se tiene la visión completa de los mensajes Inter agentes.

Anteriormente se dijo que la comunicación Inter agentes se produce entre:

- El agente supervisor de pozo y el agente de control,
- El agente supervisor de pozo y el agente de recursos,
- El agente supervisor de pozo y el agente negociador,
- Los agentes negociadores y el agente supervisor de zona,
- Los agentes negociadores de las diferentes UP's,

Tal como se muestra la Figura 38, es por ello que en los diagramas de comunicación sólo se especificarán los agentes involucrados.

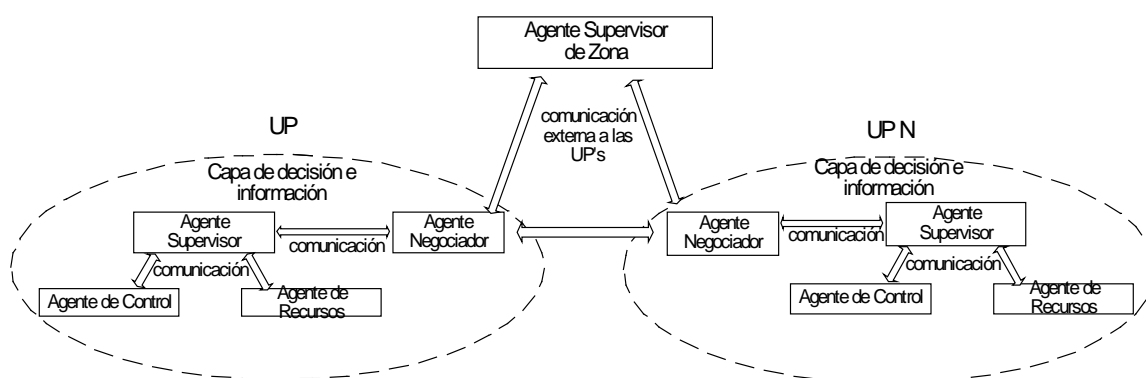
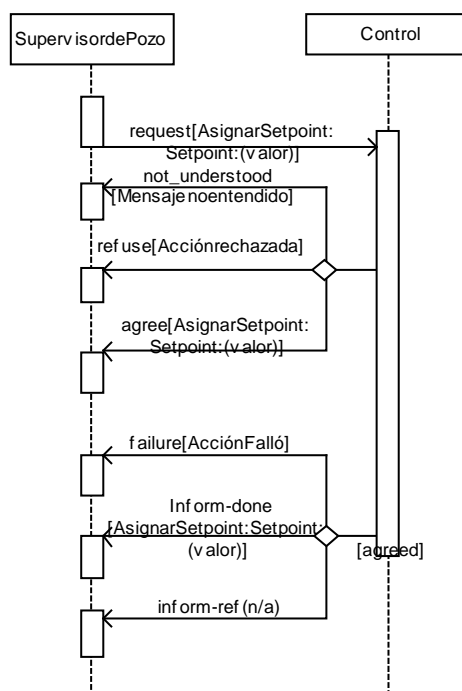


Figura 38: Comunicación inter agentes

Comunicaciones entre el agente supervisor de pozo y el agente de control.

Diagrama FIPA request protocol para la asignación de una nueva producción al nivel de planta.

La asignación de un nuevo setpoint a nivel de planta, es el producto de un acuerdo previo entre las diferentes UPs. Cada unidad de producción tiene una producción inicial fija, pero a medida que los estados de los recursos de la unidad de producción cambian, puede verse en la necesidad de cambiar esa producción inicial a una nueva, en ese caso, el agente supervisor de pozo, decide cual será la nueva producción de la UP y le solicitará al agente de control, a través del protocolo FIPA Request, que cambie el setpoint con la acción: Asignar Setpoint y le pasará el valor. Si todo esta correcto, el agente de control contestará con un “agree”, la acción solicitada y el valor del setpoint dentro del contenido del mensaje e inmediatamente enviará un “inform done”, cuando la acción se haya materializado.



Comunicaciones entre el agente supervisor de pozo y el agente de recursos.

Diagrama FIPA propose protocol para informar sobre falla en los recursos.

Cuando hay una falla en alguno de los recursos, el agente de recursos le envía al agente supervisor de pozo, a través del protocolo FIPA propose, la propuesta de la acción que va a llevar a cabo, dentro del contenido del mensaje, que será de tipo Informar Falla y el tipo de falla: *Falla grave en recursos*, si no hay energía, si se obstruyo por completo la tubería, en fin, cualquier falla grave que no permita producir nada, o será del tipo *falla en recursos* cuando la falla permita producir solo una parte de la producción asignada. El supervisor de pozo, contestará con un “agree” y la falla informada inicialmente dentro del contenido del mensaje, si está de acuerdo o con un “reject proposal” si no lo está.

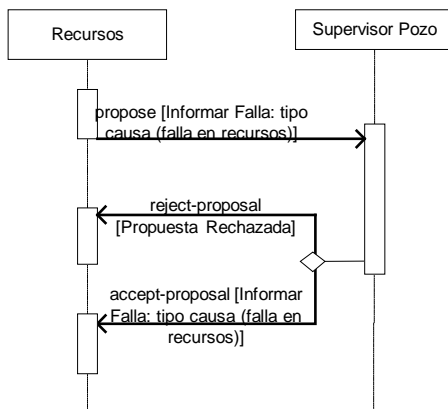


Diagrama request para solicitar información sobre los recursos

A veces es necesario que el supervisor de pozo sepa el estado actual de los recursos para analizar el estado de la UP, para negociar, etc., para ello le solicita al agente de recursos que se lo informe, enviándole a través del protocolo FIPA Request, la acción *Consultar recursos*, a lo cual el agente de recursos contestará con un “agree” si está de acuerdo y seguidamente enviará un “inform ref” y dentro del contenido del mensaje enviará el estado de los diferentes recursos consultados.

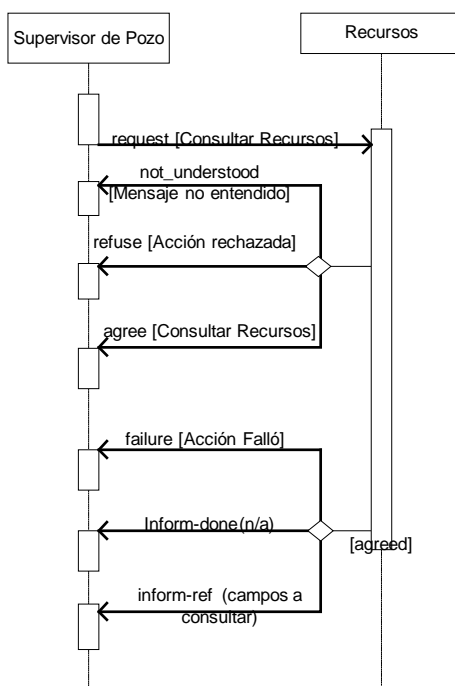
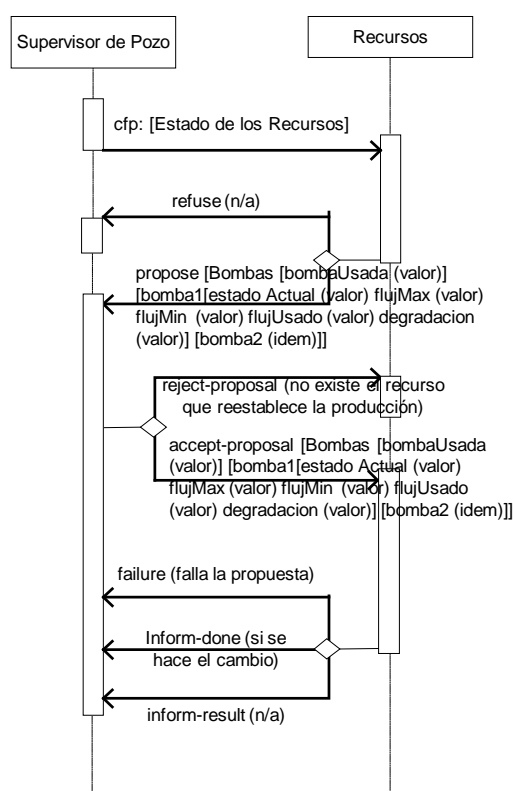


Diagrama FIPA Contract Net Protocol para solicitar cambio en las bombas

La solicitud para cambiar la velocidad o la bomba usada actualmente, podría deberse a la discrepancia entre lo que se debería producir y lo que se está produciendo, para ello, el agente supervisor inicia el protocolo de negociación FIPA Contract Net, tal que, pueda llegar a un acuerdo con el agente de recursos sobre, la bomba a usar y la velocidad. Así que, primero el agente supervisor de la UP, le pide al agente de recursos en el CFP que haga una propuesta con el estado actual de los recursos, a la cual el agente de recursos responde, después el agente supervisor estudia la propuesta y si existe alguna bomba que se pueda usar para tener la producción esperada (setpoint) acepta, de lo contrario rechaza.



Comunicaciones entre el agente supervisor de pozo y el agente negociador.

Diagrama FIPA propose para informar al agente negociador, que la UP debe negociar.

Cuando se dá alguna falla en la producción o en los recursos, que no puede ser manejada por algún agente de la UP, y por lo tanto no se puede cumplir con el objetivo de la

producción asignada, el agente supervisor envía una proposición, a través del protocolo FIPA propose, al agente negociador para informarle que la UP va a negociar, el tipo de falla y el valor de la producción faltante, es decir la que la UP no puede seguir produciendo, a lo cual el agente negociador responderá con un “agree” y la información inicial dentro del contenido del mensaje o con un “reject proposal” si no está de acuerdo.

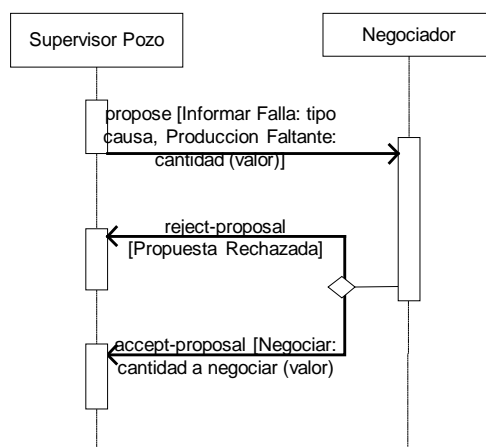
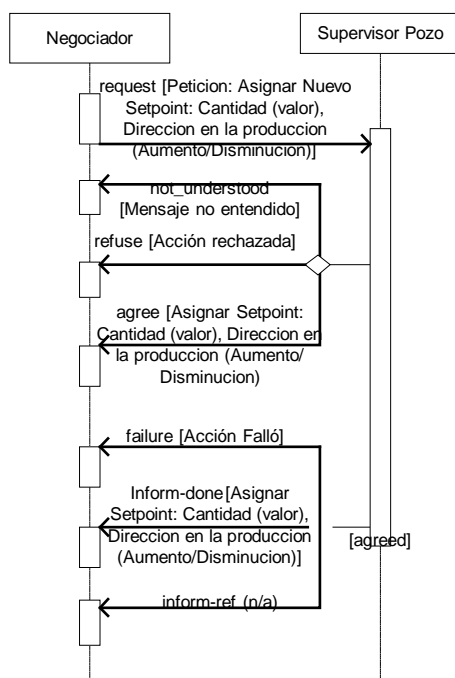


Diagrama Request para solicitar cambios en el setpoint

Después de alguna negociación con los diferentes agentes externos a la UP, podría producirse un acuerdo para cambiar el setpoint, esta acción debe ser informada al agente supervisor de pozo, para ello el agente negociador le envía al supervisor de pozo, a través del protocolo FIPA Request, la Petición: Asignar Nuevo Setpoint, el valor del nuevo setpoint y la dirección de la producción: Disminuir o Aumentar la producción, si la petición es posible, el agente supervisor de pozo contestará con un “agree” y los valores solicitados inicialmente, dentro del contenido del mensaje, de lo contrario contestará con un “reject proposal”



Comunicaciones entre los agentes externos a la UP.

Diagrama Iterated Contract Net para Negociar producción.

La producción en una UP puede ser negociada por dos razones principales: la UP ya no puede seguir produciendo la cantidad inicial que tenía asignada y, como uno de sus objetivos es cumplir con la producción de la zona, debe decidir negociar esa producción faltante, para que así las otras UPs de la zona cubran el déficit. La otra razón, es por una solicitud del agente supervisor de zona, donde pide un aumento o disminución en la producción total de la zona. Para ello, el agente iniciador, el agente negociador de alguna UP con un déficit en su producción o el agente supervisor de zona, envía a los agentes negociadores participantes un “CFP” a través del protocolo FIPA Iterated Contract Net, con la acción: *Petición*, la *cantidad a producir* (valor total a repartir entre las UPs participantes) y la *dirección de la producción* (aumentar o disminuir), dentro del contenido del mensaje, a lo cual el agente negociador de la UP participante responde con un “Propose” y dentro del contenido del mensaje una copia o replica del mensaje inicial agregando las diferentes configuraciones de las cantidades posibles de barriles por hora que la UP podría producir, si existiera alguna UP que no pudiera

producir un aumento en su producción, simplemente enviaría un “refuse”. El agente iniciador, estudia las propuestas enviadas aplicando algún método de producción escogido por el propio agente iniciador, hasta encontrar una configuración de producción para cada UP que satisfaga la cantidad solicitada inicialmente. Si esto se logra, el agente iniciador acepta las propuestas que le sirven y rechaza las propuestas restantes, de lo contrario, envía un nuevo CFP solicitándoles nuevas producciones posibles y, si todavía no logra cubrir la producción que necesita, entonces le asigna la máxima producción a cada unidad, e informa al agente supervisor de zona del déficit, si el iniciador es un agente negociador de una UP o al usuario, si el agente iniciador es el agente de zona.

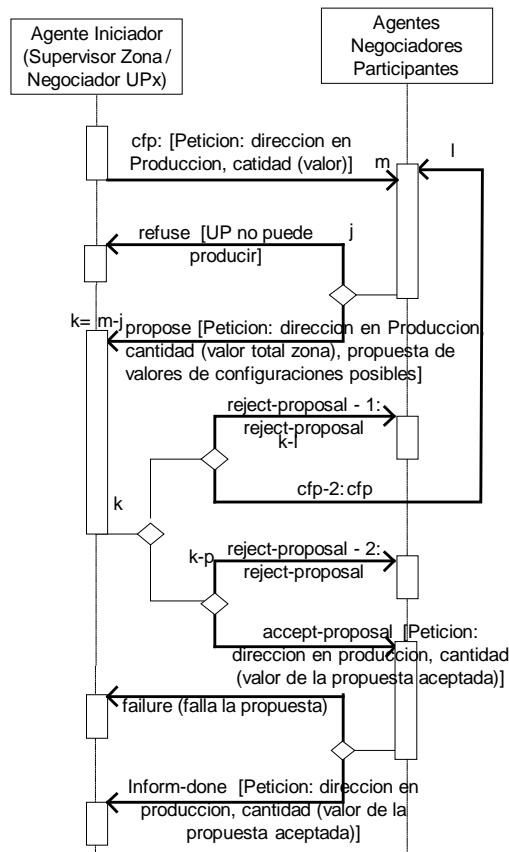
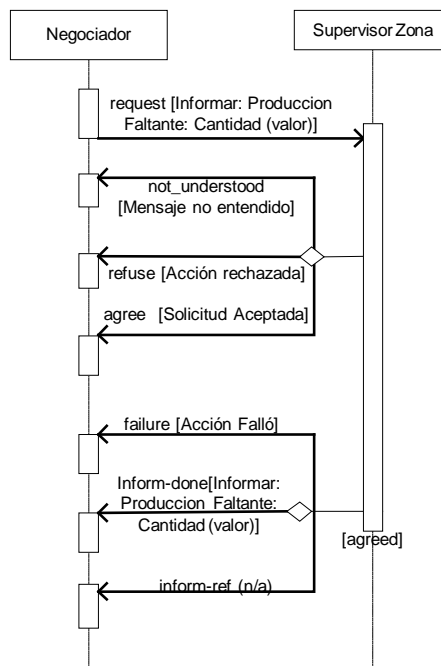


Diagrama Request para informar al agente supervisor de zona que la producción de la zona tiene un déficit.

Cuando una UP negocia con otros agentes de otras UPs parte de su producción, podría suceder que las UPs que participan en la negociación, no tengan la capacidad para producir el total de la cantidad solicitada, debido a que ya están produciendo al máximo, existe alguna falla grave en alguno de sus recursos, etc., quedando así un déficit de producción en la zona, lo cual, debe ser informado al agente supervisor encargado. Para ello, el agente negociador de la UP que no pudo negociar la producción, le informa al agente supervisor de zona, la parte de la producción que no pudo negociar, a través del protocolo FIPA Request, enviando la acción: *Informar* y el valor de la *Producción Faltante*, dentro del contenido del mensaje. Con esto el agente supervisor de zona queda informado y actualizado en cuanto a la producción real de toda la zona.



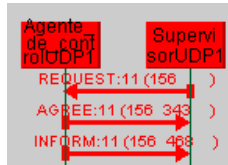
Orden del capítulo.

UDP

Clases en UML

CAPÍTULO V

ANÁLISIS DE LOS RESULTADOS



ACL Message

ACLMessage Envelope

Sender: controlUDP1@ana:1099/JADE

Receivers: SupervisorUDP1@ana:1099/JADE

Reply-to:

Communicative act: agree

Content:

```
:addresses
(sequence http://ana:7778/acc http://ana:1286/acc ht
(AsignarSetpoint
:ValorFaltante 3.4))
```

Language: fipa-sl

Encoding:

Ontology: UDP

Protocol: fipa-request

Conversation-id: C20296179_1120527621156

In-reply-to:

Reply-with: DP1@ana:1099/JADE1120527621343

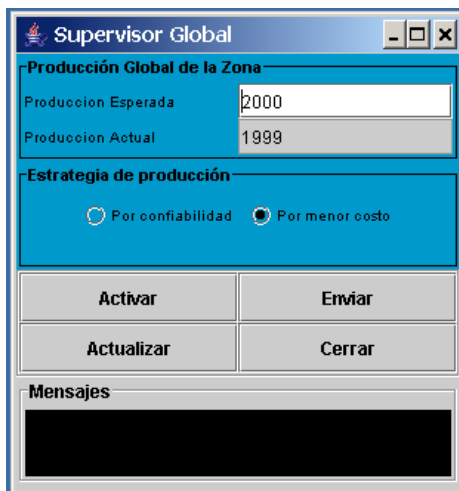
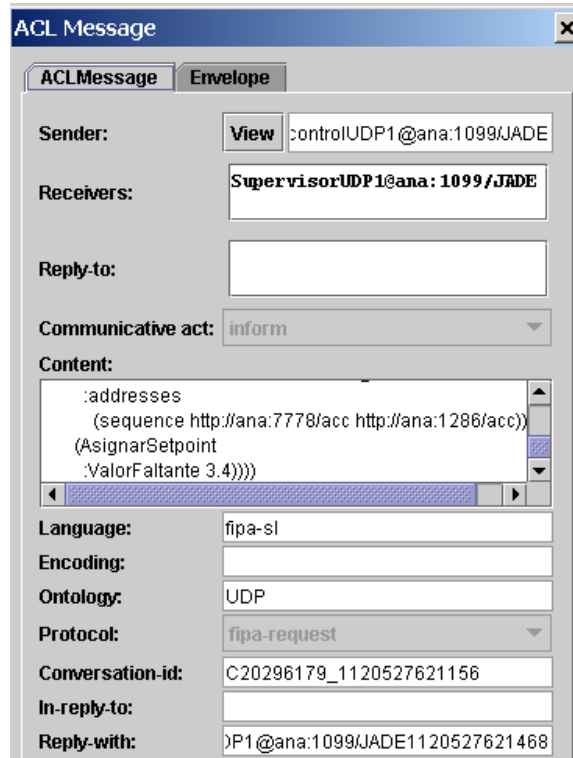


Figura 39: Interfaz Gráfica para el Agente Supervisor Global

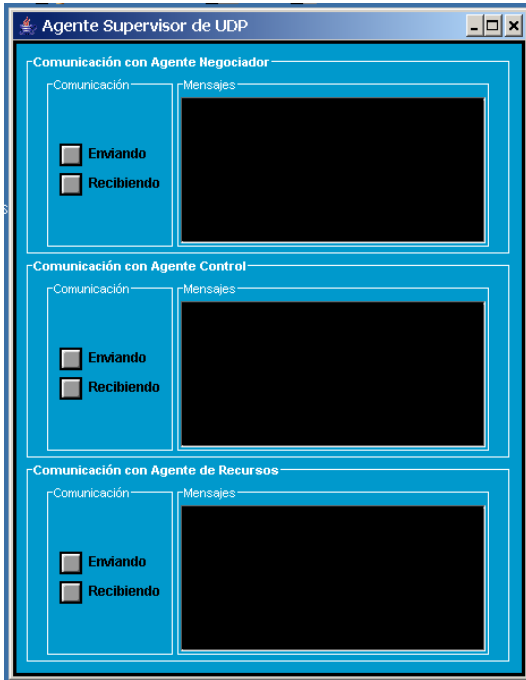


Figura 40: Interfaz Gráfica para el Agente Supervisor de la Unidad de Producción

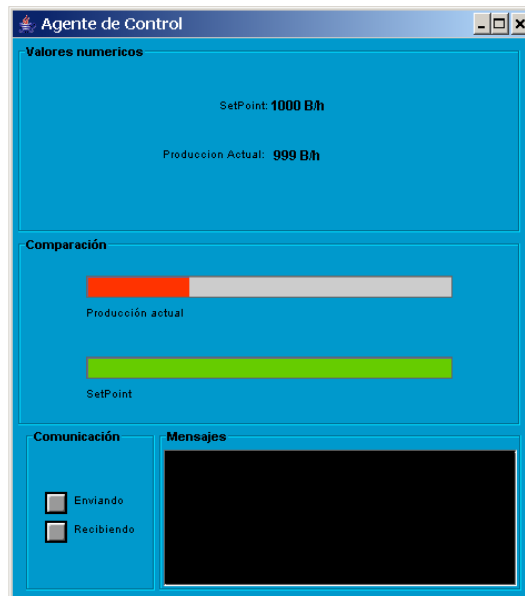


Figura 41: Interfaz Gráfica para el Agente de Control

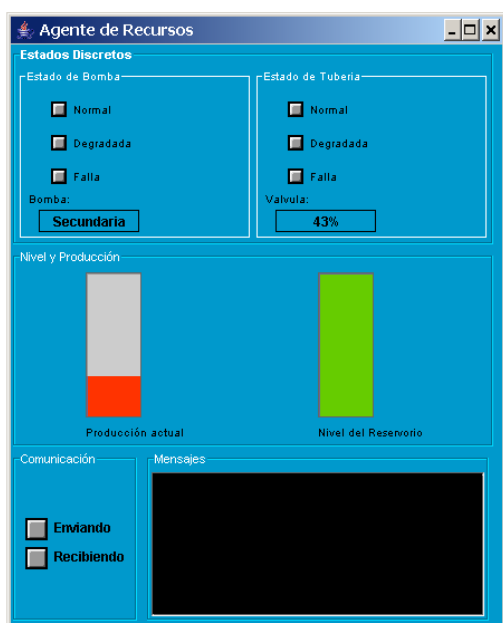


Figura 42: Interfaz Gráfica para el Agente de Recursos

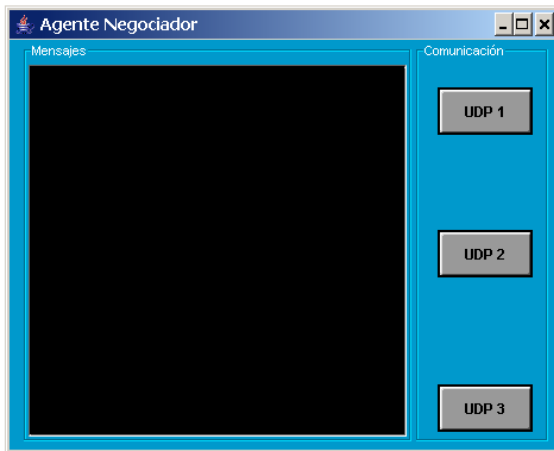


Figura 43: Interfaz Gráfica para el Agente Negociador

Capítulo V Análisis de los resultados

CAPÍTULO VI
CONCLUSIONES Y RECOMENDACIONES

Capítulo VI Conclusiones y Recomendaciones

BIBLIOGRAFÍA

Documentación general:

1. Bongaerts Luc. Integration of scheduling and control in holonic manufacturing systems. 1998. Belgium
2. Calderón Jesús. Integración de unidades de producción: una estrategia para la automatización de procesos industriales. 2001. Venezuela
3. Chacón Edgar, Besembel Isabel, Hennet Jean Claude. Coordination / Optimization of production complexes. 2002. Venezuela
4. Chacón Edgar, Besembel, Narciso, Montilva, Colina. An integration architecture for the automation of a continuous production complex. Mayo 2000. Venezuela.
5. Chacón Edgar, De Sarrazin Gisela. Automatización Integral de sistemas de producción continuos: Control y Supervisión. 1998. Venezuela.
6. Iglesias Carlos. Definición de una metodología para el desarrollo de sistemas Multiagente. 1998. España
7. Langer Gilad. HoMuCS: A methodology and architecture for Holonic Multicell. 1999. Denmark.
8. Peng Y., Finin T., Labrou Y., Cost R. S., Chu B., Long J., Tolone W. J., Boughannam A. An Agent-Based Approach for Manufacturing Integration - The CIIMPLEX Experience. USA.
9. Usher John M., Wang Yi-Chi. Negotiation between intelligent agents for manufacturing control. 2000. USA.
10. Wyns Jo. Reference Architecture For Holonic Manufacturing Systems. 1999. Bélgica.
11. Xu, Y., Brennan, R.W., Zhang, X. y Norrie, H. A Reconfigurable Concurrent Function Block Model and its Implementation in Real-Time Java. 2002. Canada.

Documentación a cerca de ACL:

1. http://www.gsi.dit.upm.es/~rgarcia/pfc_phaya/node22.html
2. http://mailweb.udlap.mx/~tesis/lis/barceinas_g_a/capitulo2.html#2.4

Documentación a cerca de JADE:

Fabio Bellifemine, Giovanni Caire, Tiziana Trucco, Giovanni Rimmasa. Jade Programmer's Guide.

Página principal de JADE:

3. <http://sharon.cselt.it/projects/jade/>

Documentación sobre KQML:

“Specification of the KQML Agent Communication Language”, The DARPA Knowledge Sharing Initiative, Esternal Interfaces Working Group, 1993.

4. <http://citeseer.nj.nec.com/finin94kqml.html>.
5. <http://www.cs.umbc.edu/kqml>.
6. <http://citeseer.nj.nec.com/finin94kqml.html>.

Documentación de FIPA:

<http://www.fipa.org>

<http://www.fipa.org/repository/componentspecs.html>.

7. FIPA Interaction Protocol Library Specification: Documento: XC00025E, año 2001, <http://www.fipa.org/specs/fipa00025/>
8. FIPA Request Interaction Protocol Specification Documento: SC00026H, año 2002, <http://www.fipa.org/specs/fipa00026/>
9. FIPA Query Interaction Protocol Specification Documento: SC00027H, año 2002, <http://www.fipa.org/specs/fipa00027/>

10. FIPA Request When Interaction Protocol Specification Documento: SC00028H, año 2002, <http://www.fipa.org/specs/fipa00028/>
11. FIPA Contract Net Interaction Protocol Specification Documento: SC00029H, año 2002, <http://www.fipa.org/specs/fipa00029/>
12. FIPA Iterated Contract Net Interaction Protocol Specification Documento: SC00030H, año 2002, <http://www.fipa.org/specs/fipa00030/>
13. FIPA English Auction Interaction Protocol Specification Documento: XC00031F, año 2001, <http://www.fipa.org/specs/fipa00031/>
14. FIPA Dutch Auction Interaction Protocol Specification Documento: XC00032F, año 2001, <http://www.fipa.org/specs/fipa00032/>
15. FIPA Brokering Interaction Protocol Specification Documento: SC00033H, año 2002, <http://www.fipa.org/specs/fipa00033/>
16. FIPA Recruiting Interaction Protocol Specification Documento: SC00034H, año 2002, <http://www.fipa.org/specs/fipa00034/>
17. FIPA Propose Interaction Protocol Specification Documento: SC00036H, año 2002, <http://www.fipa.org/specs/fipa00036/>
18. FIPA Communicative Act Library Specification Documento: SC00037J, año 2002, <http://www.fipa.org/specs/fipa00037/>

Documentación de JAVA:

<http://cariari.ucr.ac.cr/~rodolfo/java.html>

<http://es.sun.com/>

<http://www.designnews.com/article/CA86357.html>

<http://www.embedded-web.com/products/ec200.html>

Documentación de Petróleo:

<http://quipu.uni.edu.pe/OtrosWWW/webproof/acade/fipp/lucioc/Introduccion201.html>

