

Una Plataforma Multiagente para Experimentar Estrategias de Negociación

Germán Vuletich¹, Pablo Pilotti², and Ana Casali^{1,2}

¹ Facultad de Cs. Exactas, Ingeniería y Agrimensura
Universidad Nacional de Rosario (UNR)
Email: gvuletich@gmail.com

² Centro Internacional Franco Argentino de
Ciencias de la Información y de Sistemas (CIFASIS) Rosario,
Email: {pilotti, casali}@cifasis-conicet.gov.ar

Resumen La naturaleza de los sistemas multiagente hace propicia la aparición de conflictos sobre las creencias y los intereses de los agentes que interactúan en el sistema. La negociación automática es un método que se utiliza para resolver estos conflictos en sistemas multiagente. Las plataformas de negociación permiten probar y evaluar distintos comportamientos de agentes negociadores. En este trabajo, se presenta una plataforma que permite el análisis de estrategias de negociación en un sistema multiagente que utiliza el Contract Net Protocol como protocolo de negociación.

1. Introducción

La negociación automática [1] es un mecanismo para resolver conflictos en sistemas multiagente [2]. En [3], los autores distinguen tres tipos de enfoques que son utilizados para la negociación automática en entornos multiagente: los basados en *teoría de juegos*, los basados en *heurísticas* y finalmente, los basados en *argumentación*. El enfoque basado en teoría de juegos [4] estudia los modelos estratégicos de negociación utilizando como herramienta esta teoría [5]. El enfoque basado en heurísticas [6] permite relajar algunos supuestos de la teoría de juegos como la racionalidad ilimitada, a costa de una menor predicibilidad en la búsqueda de soluciones óptimas, la cual en su lugar se intenta compensar con el desarrollo de estrategias aproximadas o *heurísticas* para la obtención de soluciones *buenas* en lugar de óptimas. Estas estrategias se validan mediante pruebas y evaluación empírica simulando y comparando las propiedades de las mismas. Finalmente el enfoque basado en argumentación [3], tiene la característica de que además de las propuestas, los agentes intercambian otro tipo de información como por ejemplo críticas, justificaciones, etc. con el fin de influir en el rumbo de la negociación.

Independientemente del enfoque utilizado, gran parte de la literatura [7] acepta que la negociación automática involucra tres tópicos generales: los *Objetos de Negociación*: las cuestiones sobre las que se debe llegar a un acuerdo; el *Protocolo de Negociación*: el conjunto de normas que rigen la interacción; y los *Modelos de toma de decisiones* de los agentes (o estrategias): el mecanismo que emplean los participantes para la toma de decisiones.

Las plataformas de negociación se caracterizan por tener definido un protocolo de negociación que se utiliza durante toda la situación de negociación. Es conocido por todos los participantes y usualmente el protocolo no tiene una estrategia dominante que debería ser seguida por todos los participantes (o bien es muy costosa de calcular).

En este trabajo se propone una plataforma general de experimentación de estrategias de negociación. A diferencia de las plataformas existentes, se utiliza el Contract Net Protocol[8]. La plataforma permite generar estrategias de negociación, y analizar los comportamientos de los agentes durante una negociación de acuerdo a cada estrategia utilizada por los agentes; gracias a sus herramientas de análisis y al generador de escenarios, permite poner a prueba las estrategias que se desean evaluar y propone una estructura definida para las estrategias de negociación, de modo que sea sencillo implementarlas redefiniendo los métodos de las mismas en una clase concreta.

Este trabajo se estructura de la siguiente forma: en la Sección 2 se introducen brevemente las plataformas de negociación existentes, en la Sección 3 se presenta la plataforma propuesta, en la Sección 4 se muestra un caso de prueba. Finalmente en la Sección 5 se presentan las conclusiones y se discuten trabajos futuros.

2. Plataformas de experimentación existentes

A fines de experimentar diferentes estrategias de negociación se han desarrollado diversas plataformas de negociación. A continuación se presentan algunas de las plataformas mas relevantes.

MAGNET (<http://magnet.cs.umn.edu/>) es una plataforma de negociación generalizada para modelar subastas de múltiples agentes. Los agentes negocian y supervisan la ejecución de los contratos entre múltiples proveedores. Hay dos tipos de agente, el cliente y el proveedor. Un cliente es un agente que necesita recursos y realiza solicitudes a los proveedores, los cuales pueden contestarle con ofertas para proporcionar los recursos o servicios solicitados. **TAC** (<http://tac.sics.se/>) es una plataforma organizada como una competencia anual. En la misma cada agente participante es un agente de viajes, que tiene el objetivo de armar paquetes de viajes durante períodos de 5 días. Cada agente actúa en nombre de ocho clientes, quienes expresan sus preferencias por diversos aspectos del viaje. El objetivo del agente de viajes es maximizar la satisfacción total de sus clientes (i.e. la suma de las utilidades). Se trata de un proceso de negociación sobre múltiples características y está organizada como una negociación multilateral por medio de subastas. **GENIUS** (<http://ii.tudelft.nl/genius/>) tiene por objeto facilitar el diseño de estrategias de negociación. Se centra en la negociación bilateral y es independiente del dominio. Se supone que un determinado dominio está compuesto por una serie de asuntos a negociar y un rango de valores posibles para cada asunto. El dominio es conocido por todos los agentes y fijado durante la sesión de negociación. Un resultado de la negociación es un mapeo entre las cuestiones a negociar y los valores que éstas toman. Las interacciones entre las partes negociadoras son regulados por un protocolo de negociación que forma parte de la configuración de la sesión. GENIUS proporciona un repositorio de dominios, funciones de utilidad y un conjunto de herramientas para el análisis que calcula soluciones óptimas y muestra la evolución de

la negociación. A pesar de la variedad de posibles dominios que se pueden representar, todos ellos son de la misma clase: las preferencias son conocidas y fijas durante toda la negociación, eso es algo difícil de saber en aplicaciones del mundo real. *The Diplomacy Game* es un juego que está situado temporalmente antes de la Primera Guerra Mundial. Cada jugador está a cargo de las fuerzas armadas, organizadas en unidades, de una potencia europea importante y debe decidir en cada turno los movimientos que las distintas unidades deben ejecutar. El juego termina cuando alguien tiene un ejército lo suficientemente poderoso como para controlar la mitad de los llamados centros especiales europeos. Este es un juego de estrategia en el que los jugadores deben negociar con otros con el fin de controlar Europa. **DipGame** (<http://www.dipgame.org/>) brinda las herramientas necesarias para crear agentes que jueguen al juego.

De las plataformas mencionadas, se observa que TAC y DipGame son plataformas específicas de un dominio, mientras que MAGNET y GENIUS son más generales. MAGNET es una plataforma multiagente, pero los agentes tienen roles definidos (Clientes y Proveedores) y el mercado en lugar de ser una abstracción conceptual, se implementa como un componente definido dentro de la plataforma con el cual los agentes interactúan, en lugar de negociar directamente entre ellos. Por otro lado, MAGNET no restringe por límites de tiempo la negociación, sino que incorpora las restricciones temporales dentro de la función de utilidad, con lo cual carga las penalidades temporales en la evaluación de las ofertas. Utilizando el Contract Net protocol, como se verá en las próximas secciones, se pueden incorporar dichas restricciones temporales en la lógica de control de la negociación. Por su parte GENIUS es una plataforma general pero bilateral y que utiliza el *Alternating Offers Protocol* [9].

3. Propuesta: Plataforma General de Negociación

Se presenta una descripción general de las principales características de la plataforma propuesta y se muestra el desarrollo de un caso de uso. La plataforma permite crear un grupo homogéneo de agentes, cada uno de los cuales debe resolver un problema de optimización y desea cooperar para mejorar su situación. El deseo de cooperación queda reflejado en las propuestas y contrapropuestas que intercambian los agentes basados en sus estrategias.

El desafío que enfrentan los agentes es decidir qué propuesta y contrapropuesta realizar y cuál aceptar. Se presenta una plataforma multiagente donde se pueden crear agentes negociadores basados en heurísticas. Los agentes cuentan con un estado mental que almacena la información relevante del problema, y un mecanismo de decisión que es el encargado de generar y evaluar propuestas. Esta plataforma, implementa el Contract Net Protocol, el cual se describe a continuación.

Contract Net Protocol: El Contract Net Protocol (CNP) es un protocolo de asignación de tareas concebido en 1980 por Reid Smith[8] como un lenguaje de comunicación de alto nivel entre los nodos de un sistema distribuido. Cada nodo (agente) puede asumir en diferentes momentos el rol de Iniciador o de Participante. En la visión más general del protocolo, cuando un nodo se enfrenta a una tarea que

no puede resolver por sí solo, descompone dicha tarea en sub-tareas y las anuncia en la red de contratos. Los potenciales *contratistas* responden con ofertas y a los ganadores se les concede un contrato. A continuación se presenta una versión adaptada para esta plataforma de negociación donde un grupo de agentes desea intercambiar recursos, para mas detalles el lector puede dirigirse a la especificación FIPA³. En el protocolo hay dos tipos de agentes, el iniciador y el participante, en la Figura 1b se muestra la interacción entre los mismos. El iniciador solicita intercambios a los otros agentes mediante la emisión de una propuesta que especifica condiciones sobre los recursos a intercambiar (utilizando la función *Iniciar*, ver Definición 2). Los participantes que recibieron la propuesta son vistos como potenciales colaboradores que son capaces de generar ofertas de intercambio (mediante la función *Responder*). La ofertas de intercambio de cada participante incluye las condiciones que el participante cumplirá. Una vez que expira el plazo, el iniciador evalúa las propuestas recibidas y selecciona a los agentes para realizar la tarea (función encargada es *Evaluar*). Las ofertas de intercambio son vinculantes para los participantes, de manera que una vez que el iniciador acepta la oferta del participante, este adquiere un compromiso para llevarla a cabo. En una ronda de negociaciones, cada agente entrará a la negociación ya sea para ofrecer, solicitar u intercambiar recursos. Una negociación termina cuando todos los agentes se hayan retirado en la i -ronda ($i < n$) tras conceder un contrato, o si ningún agente logra conceder un contrato luego de n jugadas.

Entre las ventajas de CNP, además ser un protocolo de interacción por FIPA, podemos mencionar las siguientes: dado que los agentes pueden decidir ofertar o rechazar una propuesta, se obtiene un mejor dinamismo en la distribución de las tareas y mejores acuerdos ya que un agente no está forzado a negociar; balance de carga natural, dado que los agentes ocupados no necesitan ofertar; flexibilidad, dado que un agente puede entrar y salir en cualquier momento de la negociación, baja saturación en la comunicación, debido a los mecanismos de especificación que contemplan sus mensajes (direccionamiento focalizado) lo cual permite en primer lugar reducir el número de destinatarios al poder especificar los mismos dentro del mensaje, y finalmente, por la posibilidad de especificar condiciones para filtrar respuestas de participantes que no cumplen las condiciones de la propuesta.

Arquitectura de la plataforma: la arquitectura y sus componentes se muestra en la Figura 1a. A continuación se describen brevemente sus componentes principales:

- *Generador de escenarios:* Los escenarios son los datos que representan un problema de negociación. Los valores pueden ser de escenarios reales, o bien generados aleatoriamente. Los escenarios se almacenan en formato XML, en un archivo que lee cada agente al iniciar su ejecución.
- *Solver:* es utilizado para calcular soluciones de los problemas. El objetivo del diseño es representar una clase abstracta general cuyos métodos puedan ser implementados de acuerdo al problema que se esté modelando.
- *Análisis de salida:* Los valores para el análisis de la negociación son almacenados en *archivos de salida* que incluye información reelevante, como por ejemplo,

³ <http://www.fipa.org/specs/fipa00029/SC00029H.html>

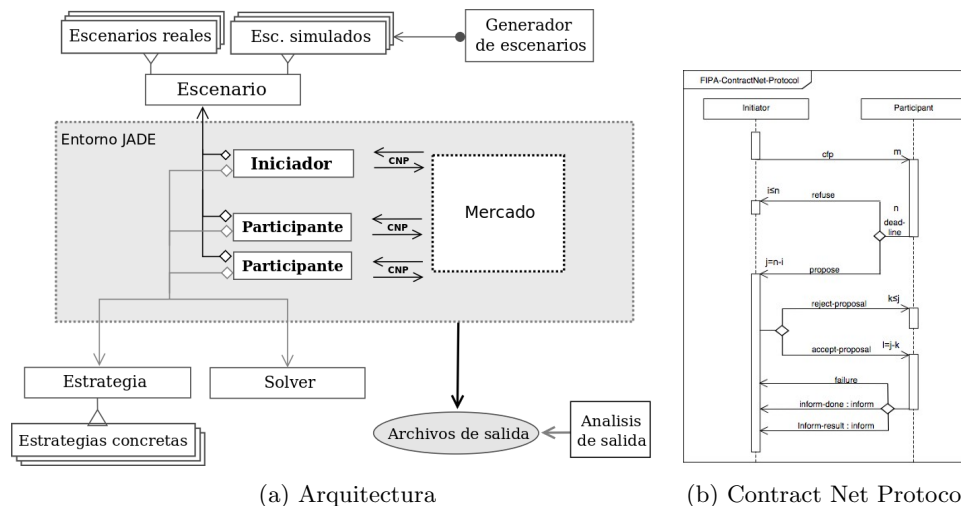


Figura 1: Plataforma de negociación basada en CNP

estrategias utilizadas, valores de los escenarios, los estados mentales de los agentes y diferentes cálculos sobre dichos datos. El objetivo es poder clasificar estos resultados de forma de poder extraer información de forma sencilla y eficiente. El componente *Análisis de salida* abarca un conjunto de utilidades para la generación de gráficos e informes a partir de los resultados de la negociación. Se busca que se puedan analizar diferentes propiedades de la negociación, como pueden ser su duración, el beneficio social (suma de los incrementos de utilidades), valores de la negociación revelados, etc.

Se implementó esta plataforma como un sistema multiagente utilizando el framework de desarrollo JADE⁴ por ser de código abierto, cumplir con las especificaciones FIPA y por que además, cuenta en sus librerías con una implementación de CNP.

4. Caso de Uso: Intercambio de Recursos entre Empresas

El mercado de un determinado sector económico, está compuesto por un conjunto finito C de empresas, cada una de las cuales produce una cantidad m de productos utilizando n recursos diferentes como materia prima. Cada empresa $\alpha \in C$ inicia su ciclo de producción con una cantidad r_i conocida de recursos por cada tipo de recurso $i \in \{1..n\}$, que es representada por un vector $r^\alpha = (r_1, \dots, r_n)$. La fabricación de cada producto le genera un beneficio a la empresa, de modo que cada empresa cuenta con un vector $u^\alpha = (u_1, \dots, u_m)$ que representa las utilidades de los m productos que fabrica. Además, cada empresa conoce cuáles y cuántos recursos son necesarios para poder fabricar cierto producto. Esta información es representada en una matriz de producción P^α de tamaño $m \times n$. Claramente cada empresa tiene que resolver un problema de programación lineal [10] para saber la cantidad de cada

⁴ <http://jade.tilab.com/>

producto que debe fabricar para obtener el beneficio máximo, suponiendo que la demanda es lo suficientemente grande. Algunas preguntas que puede plantearse cada empresa son: ¿Existen *intercambios* de recursos que mejore ese beneficio máximo? ¿Qué estrategia de negociación debería utilizar para encontrar dichos *intercambios*? ¿Qué tipo de *propuestas* debo realizar? ¿Qué tipo de *intercambios* debo aceptar?

Para experimentar estrategias de negociación, inicialmente se marcó una diferencia entre un *intercambio* y una *propuesta*. En un intercambio queda precisamente definido: qué empresas realizan el intercambio, qué recursos se intercambian y en qué cantidad. En cambio en una propuesta podrían no estar especificados todos los elementos, por ejemplo, una empresa podría proponer dar cierta cantidad de determinado tipo de recursos sin especificar que recursos desea a cambio, esperando que le hagan ofertas.

Definition 1 (Propuestas, Intercambios, Utilidad). Sean α y β dos empresas, se define:

1. O_β^α el conjunto de las propuestas de la empresa α a la empresa β formado por todos los vectores $e = (e_0, \dots, e_n)$ tal que para todo $i \in \{1, \dots, n\}$, $e_i \in \mathbb{R} \cup \{\perp\}$.
2. I_β^α el conjunto de los intercambios de la empresa α a la empresa β formado por todos los vectores $e = (e_0, \dots, e_n)$ tal que para todo $i \in \{1, \dots, n\}$, $e_i \in \mathbb{R}$.
3. La utilidad del intercambio $e \in I_\beta^\alpha$ para la empresa α es el $\max u^\alpha \times x$ sujeto a $P^\alpha \times x \leq r'^\alpha$ donde: $r'^\alpha = r^\alpha + e$ y representa al vector de recursos de la empresa α posterior al intercambio (de manera análoga para la empresa β).

Una cantidad $e_i < 0$ indica que el i -recurso es solicitado, si $e_i > 0$ indica que es ofrecido, mientras que \perp representa que la cantidad no está especificada.

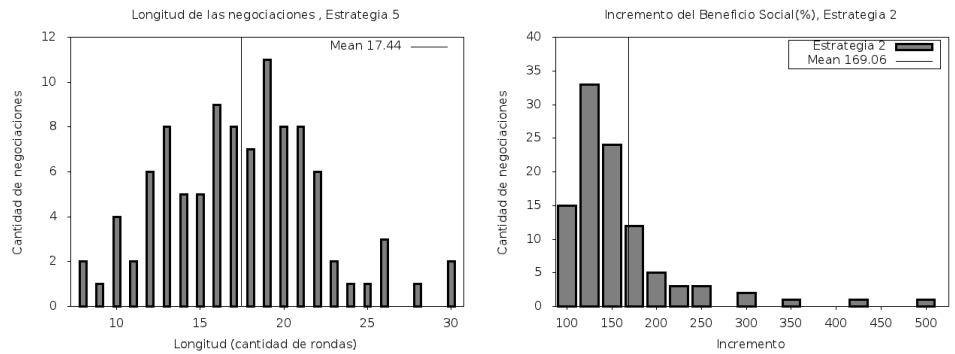
A lo largo de la negociación cada agente debe mantener la información relevante acerca del escenario de negociación y de la empresa a la que representa. Su *Estado mental* es el componente encargado de almacenar la información del problema y su *mecanismo de decisión* es el encargado de decidir cuándo y qué tipo de intercambio ofrecer y cuáles aceptar, respetando el protocolo de negociación.

Definition 2 (Agente Negociador). Un agente negociador $Ag^\alpha = \langle MS, MD \rangle$, donde MS es el estado mental y MD su mecanismo de decisión. El estado mental $MS^\alpha = \langle P^\alpha, u^\alpha, r^\alpha \rangle$ contiene la matriz de producción asociada P^α , la una función de utilidad u^α y el vector de recursos r^α , El mecanismo de decisión queda definido con una tupla de funciones $MD = \langle Iniciar, Responder, Evaluar \rangle$ donde:

- *Iniciar* : $MS \rightarrow I$, es la función que genera la propuesta inicial.
- *Responder* : $MS \times I \rightarrow I$, toma como argumento el estado mental del agente, una propuesta y devuelve una oferta de intercambio.
- *Evaluar* : $MS \times 2^I \rightarrow I$, toma como argumento el estado mental del agente, el conjunto de ofertas de intercambios recibidas y selecciona el intercambio recibido de una empresa.

Se implementaron siete estrategias de negociación, definiendo de manera concreta las funciones *Iniciar*, *Responder*, *Evaluar*. Las funciones utilizan el componente *Solver* para tomar decisiones, En este caso se utilizó el solver *lpsolve* (sourceforge.net/projects/lpsolve/). Luego se utilizó el componente *Generador de Escenarios* para crear de manera aleatoria 100 estados mentales iniciales. Se fijó el rango

[10...1000] para la generación aleatoria de cantidades de recursos en cada escenario y los coeficientes de las funciones de utilidad se asignaron sobre valores del rango [10...100], con matrices de producción de 4×5 (i.e., 4 recursos y 5 productos fabricados por empresa). Los coeficientes de incidencia de cada recurso en un producto en particular (i.e., la cantidad de cada recurso para fabricar un producto) variaron aleatoriamente entre 0 y 100. Finalmente se simularon negociaciones con 5 agentes. Para cada simulación, todos los agentes utilizaron la misma estrategia. Los archivos de salida fueron tomados por el componente *Análisis de salida* que



(a) Estrat. 5: Longitud de la negociación (b) Estrat. 2: Incremento del beneficio social

Figura 2: Comportamiento global

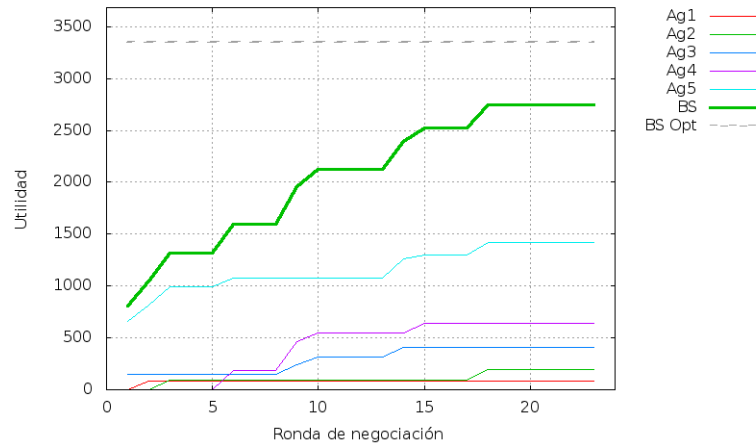


Figura 3: Comportamiento particular: Estr.6, en escenario Nro 42

automáticamente generó gráficos sobre el conjunto de casos. En la Figura 2a se muestra la distribución de la *Longitud de la negociación* de la Estrategia 5, vemos que el promedio de mensajes intercambiados es de 17.44 mensajes. En la Figura 2b

se observa el porcentaje en el que aumentó el beneficio social con la negociación en la Estrategia 2. La función de beneficio social utilizada es la suma de las utilidades. Además, el componente *Análisis de salida* genera gráficos para observar el comportamiento de un escenario en particular por ejemplo, en la Figura 3 se observa el incremento de la utilidad de cada agente a medida que avanza la negociación en el caso de estudio número 42. En esta figura puede observarse el incremento del beneficio social a medida que avanza la negociación y el beneficio social óptimo.

5. Conclusiones y Trabajos Futuros

Se ha presentado una plataforma multiagente general para la experimentación de estrategias de negociación y se ha probado con un caso de uso que puede extraerse información relevante para el análisis de distintas estrategias. Este análisis puede evaluarse en función de la duración de la negociación, beneficio social, incremento en las utilidades de un agente, etc. Consideramos que este framework tiene gran potencial de desarrollo futuro. Se planea mejorarlo en varias direcciones. Primero, formalizar una interfaz a distintos problemas de optimización para poder abarcar la mayor cantidad de situaciones de negociación posibles. En segundo lugar, se desea generalizar la plataforma para que soporte versiones extendidas del CNP para que los agentes argumenten [11].

Además, se pretende que los agentes puedan correr en sistemas operativos de los dispositivos móviles (e.g. Android, iOS).

Referencias

1. Fatima, S., Kraus, S., Wooldridge, M.: Principles of Automated Negotiation. Cambridge University Press (2014)
2. Wooldridge, M.: Introduction to Multiagent Systems. John Wiley & Sons, Inc., New York, NY, USA (2001)
3. Rahwan, I., Ramchurn, S.D., Jennings, N.R., Mcburney, P., Parsons, S., Sonenberg, L.: Argumentation-based negotiation. *Knowl. Eng. Rev.* **18** (December 2003) 343–375
4. Osborne, M.J., Rubinstein, A.: A Course in Game Theory. Volume 1 of MIT Press Books. The MIT Press (December 1994)
5. Rosenschein, J.S., Zlotkin, G.: Rules of Encounter - Designing Conventions for Automated Negotiation among Computers. MIT Press (1994)
6. Faratin, P., Sierra, C., Jennings, N.R.: Negotiation decision functions for autonomous agents. *Robotics and Autonomous Systems* **24**(3–4) (1998) 159 – 182 Multi-Agent Rationality.
7. Jennings, N.R., Faratin, P., Lomuscio, A.R., Parsons, S., Sierra, C., Wooldridge, M.: Automated negotiation: Prospects, methods and challenges. *International Journal of Group Decision and Negotiation* **10**(2) (2001) 199–215
8. Smith, R.G.: The contract net protocol: High-level communication and control in a distributed problem solver. *IEEE Trans. Comput.* **29**(12) (December 1980) 1104–1113
9. Rubinstein, A.: Perfect equilibrium in a bargaining model. *Econometrica* **50**(1) (1982) pp. 97–109
10. Chvatal, V.: Linear Programming. Series of books in the mathematical sciences. W. H. Freeman (1983)
11. Ojha, A.C., Pradhan, S.K., Patra, M.R.: Extending contract net protocol for arguing agents. In: IICAI. (2007) 1354–1365