Designing Responsive and Deliberative Automated Negotiators

P. Faratin *
QMW
University of London
P.Faratin@qmw.ac.uk

C. Sierra
IIIA (CSIC), Spain.
Sierra@iiia.csic.es

N.R. Jennings
QMW
University of London
N.R.Jennings@qmw.ac.uk

P. Buckle
Nortel Networks, UK.
Pbuckle@nortelnetworks.com

Abstract

This paper presents a multi-issue negotiation model that can be used for guiding agents during distributed problem solving. The model is composed of protocols that govern and manage agent interactions, and an agent architecture that represents decision mechanisms that assist agents during their negotiations.

keywords: Automated Negotiation, Agent Architecture

Introduction

Automated agents are autonomous entities that decide for themselves what, when, and under what conditions their actions should be performed. Since agents have no direct control over one another, they must persuade others to act in a particular manner. The type of persuasion we consider in this paper is negotiation which we define as *a process by which a joint decision is made by two or more parties. The parties first verbalise contradictory demands and then move towards agreements* (Pruitt 1981). Furthermore, negotiating agents may populate different types of environments that require either very simple and responsive decisions to be made (e.g. buying and selling goods in an auction), or complex and deliberative problem solving activities (e.g. planning), or a combination of the two. Therefore, we view negotiation decisions as being composed of responsive and/or deliberative components. The outcome of these decisions can result in either concession or search for new alternatives.

Traditionally, formal models of choice achieve coordination through a priori specification of the negotiation space: the issues that agents negotiate over and their possible values that determine the set of alternative solutions. Negotiation is then considered as an optimisation problem where, given the utility function of all the agents, the best solution is obtained. This methodology is often adopted in classical Game Theory. However, such formal models of choice often ignore *interactions* and involve unrealistic assumptions (such as common knowledge and an unlimited computational ability). In such models interactions are viewed as unnecessary since rational and super-logical agents can reach agreements instantly given the common knowledge and unlimited computational power assumptions.

* Supported by a grant from Nortel Networks

An alternative coordination methodology is to specify the rules of interaction — *who* can say *what* and *when* (the absence of any normative rules of behaviour can lead to chaotic interactions). In this paper we follow this second approach, and ensure coordinated behaviour by defining our protocol as an extension of the normative rules of the Contract Net Protocol (Davis & Smith 1988).

In addition to a protocol of interaction, agents need the capability to represent and reason about, within their information and resource bounds, both their internal and their external world and with the capacity to interact according to the above protocol. It is this individual agent modelling that we focus on in this paper. Our design choices have been strongly influenced by two major application developments with which we have been involved. These are the ADEPT system for business process management,(Sierra, Faratin, & Jennings 1997), and the *Foundation of Intelligent Physical Agents* (FIPA) field trial of in the telecommunication domain,(FIPA97 1997). Here we focus exclusively on the latter scenario.

This paper extends our previous work [(Faratin, Sierra, & Jennings 1998)] on negotiation models in the following ways. Firstly, the agent architecture has been updated from a purely responsive mechanism to include higher level deliberative mechanisms (involving the generation of trade offs and the manipulation of the set of issues under negotiation). Secondly, the negotiation protocol has been expanded to account for these new mechanisms. More generally speaking, this paper advances the state of the art in automated negotiation by designing components of a negotiation architecture that allows agents to be both responsive and deliberative. This, in turn, enables them to participate in more varied types of negotiation processes.

The example scenario is introduced first, followed by the developed negotiation protocol. Next the individual agent architecture describes the evaluation and offer generation mechanisms. Our model is then compared with other negotiation models. Finally, we present the conclusions reached and future avenues of research.

FIPA's Negotiation Scenario

The scenario is based on the use of negotiation to coordinate the dynamic provisioning of resources for a Vir-

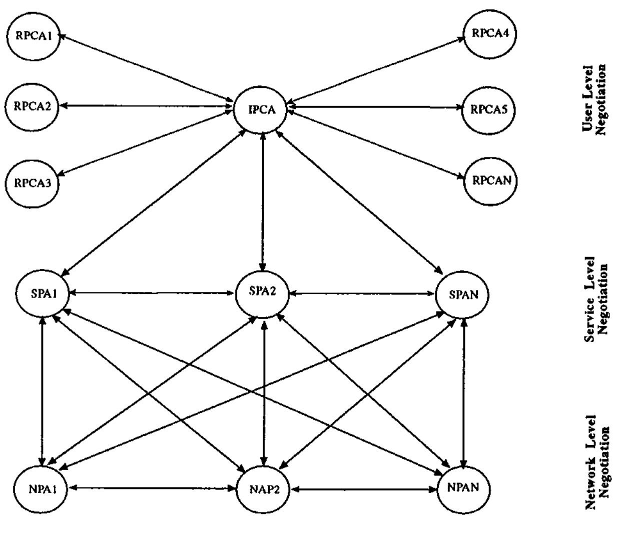

Figure 1: FIPAs VPN Provisioning Scenario.

tual Private Network (VPN) used for meeting scheduling by end users.[1] This service is provided to the users by service and network providers. The scenario is composed of a number of agents that represent the users, the service and network providers (see figure).

Individuals using the system are represented by user agents that are collectively refered to as Personal Communication Agents or PCAs. PCA agents are composed of *IPCA* and *RPCAs*; the Initiating *PCA*, represents the user who wants to initiate the meeting, and the Receiving *PCA/s*, that represent the party/parties that are required to attend the meeting. Interactions between PCAs can be multilateral (involving one *IPCA* and multiple *RPCAs*) and are centred around negotiation over meeting scheduling. Each agent negotiates on behalf of their user and has the goal of establishing the most appropriate time and security level (see below) for the service requested by the *IPCA*. The set of issues over which PCAs negotiate are: $\{Service\_Type, Security, Price, Start\_Time, Duration\}$. *Service_Type* denotes the choice of the service (eg. video, audio or mixture of both). *Price* is the share of the price the agents should pay for the service. *Start_Time* and *Duration* are the time the service will commence and its length, respectively. *Security* encodes the privacy of the meeting and is represented by both the method of security (e.g. in the order of value to PCAs: Entrust, Verisign or Mircosof) and the level of the security method (again in the order of value: confidentiality, integrity and authentication).

The requirements of the *IPCA* and the *RPCAs* are constrained by what resources are available at the network level. For example, the network may be heavily loaded at the time the service is required by the *PCAs*.

[1] A VPN refers to the use of a public network (such as the Internet) in a private manner.

Since the network is only visible to *IPCA* through the Service Provider Agents (*SPAs*), the threads of *IPCA* and *RPCAs* negotiation are executed in parallel with negotiation between *IPCA* and *SPAs*. Note however that the interactions between *IPCA* and *SPA* directly influence the meeting scheduling negotiations between *IPCA* and *RPCAs*. Furthermore, we assume only bilateral negotiation between *IPCA* and *SPAs*. However, each *SPA* can make agreements with *IPCA* for services and then outsource these commitments by initiating negotiation with other *SPAs*. The set of issues in the negotiation between *IPCA* and *SPAs* is the same as that between *IPCA* and *RPCAs* except there is the additional element *Participants* that is the list of users (represented by *RPCAs*) specified to be included in the requested service.

Either concurrently or after the service is provisioned between *IPCA* and *SPA*, multiple threads of negotiation are initiated between the *SPA* and the Network Provider Agents, *NPAs*, that manage the infrastructure and low level aspects of the IP network. These threads of interaction are multilateral since each *NPA* manages only a subset of the IP network. Therefore, the *SPA* must negotiate with a number of *NPAs* in order to secure resources for the services it provides to *IPCA*. The set of issues in the thread of negotiation between *SPA* and *NPAs* includes: $[Quality\_of\_Service, Security, Participants, Price, Start\_Time, Duration]$. Here *Quality_of_Service*, or *QoS*, represents the "goodness" of the service from an agent's perspective. *QoS* is often discussed as if it were composed of a number of sub issues such as, the *Bandwidth* (capacity of the link), the *latency* (delay imposed by the network on packets), the *jitter* (maximum time deviation acceptable during transmission), the *availability* (percentage of time over which the service is required) and *packetloss* (percentage of total packets lost during lifetime of the provisioned service). Therefore, the *QoS* issue is represented as a set of sub issues.

Features of the Scenario

Negotiation, in the scenario above, exhibits the following characteristics:

- Agents negotiate over *services*. Services have a number of features/issues associated with them (e.g. *Price*, *Duration* etc.), some of which can be dynamically introduced and retracted (eg. *QoS*). Successful negotiation involves resolution of all these issues to the satisfaction of all the parties involved.
- Since agents are autonomous, the factors that influence their negotiation stance and behaviour are private and not available to other parties. Thus, agents do not know what utilities their opponents place on various outcomes, what reasoning models they employ, their opponent's constraints or whether an agreement is even possible at the outset (i.e. participants may have non-intersecting ranges of accept-

ability).

- Since plans and execution of services/activities are real time and dependent on one another, the provisioning process should respect the time and resource levels of the agents. Negotiation should therefore be responsive to the time and resource levels of the agent. For example, if the environment can afford it (in terms of time, resources, etc.) then an agent may decide to engage in complex deliberation procedures involving a more refined search of the space of possible outcomes. For instance, *SPAs* and *NPAs* can engage in costly computation and selection procedures for contracts that manipulate or trade off the set of issues involved in negotiation. Alternatively, as the environment changes (deadline to reach an agreement is approaching fast, resource usage for negotiation has reached some critical level, the other agent is exhibiting a reluctance to change its offer, etc.) then one/both of the agents may begin to adopt a more responsive attitude towards their environment by conceding. Thus responsive negotiation behaviour is similar to reactive behaviours that considers environmental conditions and are simple and uncostly responses to its changes.

The Negotiation Protocol

Coordinated behaviour during negotiation is enforced through the normative rules of the negotiation protocol. Here we restrict ourselves to bi-lateral negotiations (however, multi-lateral negotiations can be shown to be equivalent to a series of bi-lateral negotiations (Binmore 1985)).

The protocol (figure 2) starts by a dialogue to establish the conditions for negotiation (deadline, initial issues, etc.). Then, one of the agents makes an offer (transition from state 1 to state 2 or 3) for contract ϕ. After that, the other agent can make a counter-offer or a tradeoff (moving to state 2 or 3 depending on who started), and the agent that started the negotiation can in turn make a new counter-offer or a new tradeoff (going back to state 2 or 3). Since information models of agents are not publicly known, offers may be outside the mutual zone of agreement. Therefore, agents may iterate between states 2 and 3, taking turns to offer new contracts. In either of these two states, one of the agents may accept the last offer made by the opponent (moving to state 4) or withdraw from the negotiation (moving to state 5). Agents withdraw from the negotiation process when the deadline of negotiation has been reached.

While in state 2 or 3, agents may start an ellucidatory dialogue to establish a new set of issues to negotiate over. This protocol is a natural extension of the contract net protocol permitting iterated offer and counter-offer generation and permitting the modification of the set of issues under negotiation. Although we cannot guarantee termination or convergence in the general case, in practice the existence of time deadlines ensures that the protocol terminate in most practical cases.

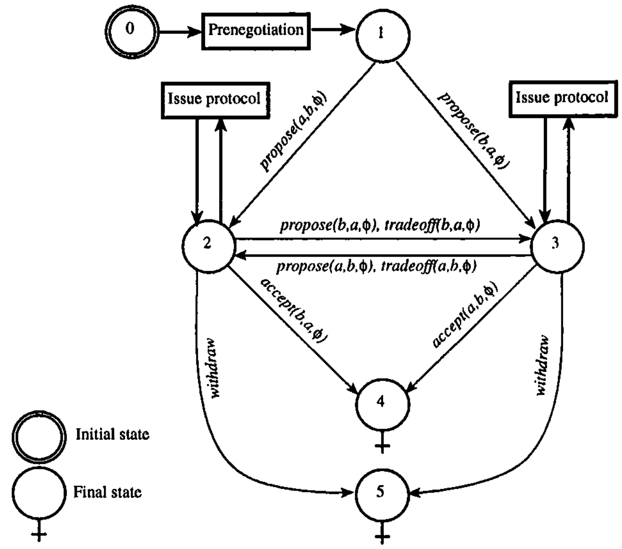

Figure 2: Negotiation protocol.

Agent Negotiation Architecture

The main contribution of the research reported here is the specification of an architecture that structures the individual agent's reasoning throughout the negotiation problem solving. An agent's reasoning process is assumed to be constrained by maximisation of some value function (Raiffa 1982). Given this rationality stance, the decisions faced by agents involved in a negotiation process are often a combination of: *offer generation decisions* (what initial offer should be generated? what counter offer should be given in situations where the opponent's offer is unacceptable?), and *evaluatory decisions* (when should negotiation be abandoned? and when should an agreement be deemed to have been reached?). The solution to these decision problems is captured in the agent architecture. The components (or what is referred to as the mechanisms) of the architecture that is responsible for generation of offers and counter offers are based on a distinction between mechanisms that are computationally uncostly and are responsive to the environment, and mechanisms which are relatively more costly because they engage in a more sophisticated search of the solution space.

The mechanism that assists an agent with its evaluation of offers is described first, followed by the generation mechanisms. Since the main novelty of the work reported here is the deliberative mechanism, the reader is referred to (Faratin, Sierra, & Jennings 1998) for an indepth explanation of the evaluatory and responsive mechanisms.

Evaluation Mechanism

The evaluation process involves computing the value/score of a proposal or a contract. When an agent a receives an offer x from b at time t, $x^t_{b\to a}$, over a set of issues J, ($x = (x[j_1], \ldots, x[j_n])$ where $j_i \in J$), it rates the overall contract value using the following weighted linear additive scoring function:

$$V^a(x) = \sum_{1\leq i\leq n} w^a_{j_i} V^a_{j_i}(x[j_i])$$

where $w^a_{j_i}$ is the importance (or weight) of issue j_i such that $\sum_{1\leq i\leq n} w^a_{j_i} = 1$. Given that the set of negotiation issues can dynamically change, agents need to dynamically change the values of the weights. The score of value $x[j]$ for agent a, given the domain of acceptable values $\mathcal{D}_j$, is modelled as a scoring function $V^a_j : \mathcal{D}_j \to [0, 1]$. For convenience, scores are bounded to the interval $[0, 1]$ and the scoring functions are monotonous for quantitative issues. Note that in the above formulation we assume scores of issues are independent.

Given the score of the offered contract, the evaluation function will determine whether to accept or reject the contract or whether to generate a new contract to propose back to the other agent. The mechanisms that generate new contracts are presented in the sections below.

Responsive Mechanisms

Responsive mechanisms model reactive behaviours to a number of environmental factors. The underlying rationale and motivation for the design of these mechanisms has been the need to model concessionary behaviours which are initiated by progressivley more important environmental factors during the course of negotiation process. For example, if *IPCA* has committed many resources during its negotiation with *SPA* and the time of the required video service with other *RPCAs* is soon, then the *IPCA* may prefer simple and less costly decision mechanisms that can result in concessions. Concession may result in an agreement, and therefore not only free *IPCA's* resources,which can be used for other activities, but also achieve the goal of establishing a meeting with the *RPCAs*.

Responsive mechanisms generate offers by linearly combining simple decay functions, called *tactics*. Tactics generate values for issues using only a single environmental criterion. We have designed three families of tactics:

- **Time-dependent tactics**: These model increasing levels of concession as the deadline for the negotiation approaches.
- **Resource-dependent tactics**: These model increasing levels of concession with diminishing levels of resources.
- **Behaviour-dependent tactics**: Concession here is based on the concessions of the other negotiating party.

To determine the best course of action, an agent may need to consider and assess more than just one environmental condition. Since each tactic generates a value for an issue using only a single criterion, the concept of *strategy* is introduced to model the modification, over time, of tactic weights as the criteria change their relative importance in response to environmental changes.

Deliberative Mechanisms

Our model contains two kinds of deliberative mechanism: *trade off* and *issue set manipulation.*

Trade Offs: A trade off is where one party lowers its score on some issues and simultaneously demands more on other issues. For example, for the *IPCA* offering a lower *Price* for a later *Start_Time* of may be equivalent in value (depending on the weights of the two issues) to offering a higher *Price* for an earlier *Start_Time*. However, this change in offer *may* benefit the *SPA* agent.[2] Thus, a trade off is a search for a contract that is equally valuable to the previously offered contract, but which may be of greater benefit the other party.

This decision making mechanism is more costly than the responsive mechanism because it involves searching all, or a subset of, possible contracts with the same score as the previously offered contract and then selecting the contract that is the "closest" to the opponent's last offer. The search is initiated by first generating new contracts that lie on what is called the iso-value (or indifference) curves (Raiffa 1982). Because all the potential contracts lie on the same iso-value curve the agent is indifferent between them. More formally, an iso-curve is defined as:

Definition 1 *Given a scoring value θ, the iso-curve set at degree θ for agent a is defined as:*

$$iso_a(\theta) = \{x \mid V^a(x) = \theta\}$$

The selection of which contract to offer is then modelled as a "closeness function". The theory of fuzzy similarity can be used to model "closeness". The best trade off is the one that is most similar contract on the iso-curve. A trade off can now be defined as:

Definition 2 *Given an offer, x, from agent a to b, and a subsequent counter offer, y, from agent b to a with $\theta = V^a(x)$, a trade off for agent a with respect to y is defined as:*

$$tradeoff_a(x, y) = \arg\max_{z\in iso_a(\theta)} \{Sim(z, y)\}$$

[2] This evaluation is uncertain since information models are private—*IPCA* does not know the valuation methodology or the importance *SPA* attaches to the issues in negotiation.

where the similarity, Sim, between two contracts is defined as a weighted combination of the similarity of the issues:

Definition 3 *The similarity between two contracts x and y over the set of issues J is defined as:*

$$Sim(x,y) = \sum_{j \in J} w_j^a Sim_j(x[j], y[j])$$

With, $\sum_{j \in J} w_j^a = 1$. Sim_j is the similarity function for issue j.

Following the results from (Valverde 1985), a similarity function, that is, a function that satisfies the axioms of reflexivity, symmetry, and t-norm transitivity, can always be defined as a conjunction (modelled as the infimum) of appropriate fuzzy equivalence relations induced by a set of criteria functions h_i. A criteria function is a function that maps from a given domain into values in $[0,1]$. For example, a function that models the criteria of whether a price is low, $lowprice : Price \to [0,1]$, could be:

$$lowprice(x) = \begin{cases} 1 & \text{x} < \text{£}10 \\ \frac{\pounds 20 - x}{\pounds 10} & \text{£}10 < \text{x} < \text{£}20 \\ 0 & x \geq \pounds 20 \end{cases}$$

Hence the similarity between two values for issue j, $Sim_j(x,y)$ is defined as:

Definition 4 *Given a domain of values D_j, the similarity between two values $x, y \in D_j$ is defined as:*

$$Sim_j(x,y) = \min_{1 \leq i \leq m} (h_i(x) \leftrightarrow h_i(y))$$

where $\{h_1, \ldots, h_m\}$ is a set of comparison criteria with $h_i : D_j \to [0,1]$, and $\leftrightarrow$ is an equivalence operator. Simple examples of the equivalence operator, $\leftrightarrow$, are $h(x) \leftrightarrow h(y) = 1 - \mid h(x) - h(y) \mid$ or $h(x) \leftrightarrow h(y) = min(h(y)/h(x), h(x)/h(y))$.

Issue Set Manipulation: Our other deliberation mechanism is the issue set manipulation. Negotiation processes are directed and centred around the resolution of conflicts over a set of issues J. This set may consist of one or more issues (distributed and integrative bargaining respectively, (Raiffa 1982)). For simplification we assume the ontology of the set of possible negotiation issues J, is shared knowledge amongst the agents. It is further assumed that agents begin negotiation with a prespecified set of "core" issues, $J^{core} \subseteq J$, and possibly other mutually agreed non-core set members, $J^{\neg core} \subseteq J$. Alterations to J^{core} is not permitted since some features such as the *Price* of services are mandatory. However, elements of $J^{\neg core}$ can be altered dynamically. Agents can add or remove issues into $J^{\neg core}$ as they search for new possible and up to now unconsidered solutions. In the VPN scenario above, agents negotiate over core issues. The negotiation between *SPAs* and *NPAs* consists of offers over non-core issues. For example, a *SPA* may begin *QoS* negotiation with a *NPA* specifying only *Bandwidth*. However, subsequently *NPA* may decide to include into the *QoS* negotiation a *packetloss* issue with a high value if *SPA* has demanded a high capacity *Bandwidth*. Alternatively, *SPA* may decide to remove the *Bandwidth* issue from the *QoS* negotiation with *NPA* if *IPCA* has changed its demand from a high quality video service to a standard audio service.

If J^t is the set of issues being used at time t (where $J^t = \{j_1, \ldots, j_n\}$), $J - J^t$ is the set of issues not being used at time t, and $x^t = (x[j_1], \ldots, x[j_n])$ is a's current offer to b at time t, then issue set manipulations is defined through two operators: *add* and *remove*.

The *add* operator assists the agent in selecting an issue j' from $J - J^t$, and an associated value $x[j']$, that gives the highest score to the agent.

Definition 5 *The best issue to add to the set J^t is defined as:*

$$add(J^t) = \arg \max_{j \in J - J^t} \{ \max_{x[j] \in D_j} V^a(x^t.x[j]) \}$$

where . stands for concatenation.

An issue's score evaluation is also used to define the *remove* operator in a similar fashion. This operator assists the agent in selecting the best issue to remove from the current negotiation set J^t.

Definition 6 *The best issue to remove from the set J^t (from a's perspective), is defined as:*

$$remove(J^t) = \arg \max_{j_i \in J^t - J^{core}} \{V^a(x)\}$$

with $x = (x^t[j_1], \ldots, x^t[j_{i-1}], x^t[j_{i+1}], x^t[j_n])$

The *remove* operator can also be defined in terms of the aforementioned similarity function. This type of similarity-based *remove* operator selects from two given offers x, from agent a to b, and y, from agent b to a, which issue to remove in order to maximise the similarity between x and y. Therefore, compared to the previous *remove* operator, this mechanism can be considered as more cooperative:

Definition 7 *The best issue to remove from a's perspective from the set J^t is defined as:*

$$\begin{aligned} remove(J^t) \quad = \quad & \arg \max_{j_i \in J^t - J^{core}} \{ sim(\\ & (x[j_1], \ldots, x[j_{i-1}], x[j_{i+1}], \ldots, x[j_n]), \\ & (y[j_1], \ldots, y[j_{i-1}], y[j_{i+1}], \ldots, y[j_n])) \} \end{aligned}$$

It is not possible to define a similarity-based *add* operator since the introduction of an issue does not permit an agent to make comparisons with the opponent's last offer (simply because there is no value offered over that issue).

Agents deliberate over how to combine these *add* and *remove* operators in a manner that maximises some measure — such as the contract score. However, a search of the tree of possible operators to find the optimum set of issues may be computationally expensive.

To overcome this problem we intend to implement any-time algorithms and use the negotiation time limits to compute a, possibly sub-optimal, solution. Another computational requirement of these mechanisms is the need for an agent to dynamically recompute the issue weights.

The protocol for establishing a new set of negotiating issues is isomorphic to the negotiation protocol described in figure 2. The pre-negotiation phase is omitted (since the current set of issues has already been agreed). ϕ is replaced by a new set of issues S, and primitives *propose* and *tradeoff* are replaced by *newset* —a request for a new set of issues to be included in to the negotiation. Each negotiating agent can start a dialogue over a new set of issues S (state 1 to 2 or 3). Each agent can then either propose a new set (transition from state 2 to 3, depending on who started the dialogue), accept the other's proposed set (state 4) or withdraw and continue with the original set (state 5).

Related Work

Because negotiation is prevalant and important in many types of interactions it has been studied in a number of related disciplines. The central focus of the work reported here has been the design of a negotiation agent architecture for structured interactions for services in real-world environments. Our work is closely related to the Contract Net Protocol (Davis & Smith 1988), where a protocol is used for modelling interactions. However, unlike the CNP we do not assume agents are cooperative. Furthermore, because of the privacy of information models the search for acceptable solutions may be more elaborate than the CNP's two messages — negotiation is an iterative process. In addition to this, CNP is a theory of system architecture (theory specifies behaviours at the level of interaction protocol) and is silent with respect to the individual agent architecture. Consequently, like game theory, it is inadequate for agent design since any agent architecture is as good as any other as long as they obey the protocol specification. In contrast, our model not only specifies a negotiation protocol used for iterative interaction modelling, but it also provides both responsive and deliberative mechanisms that agents can implement and execute according to their own requirements.

Iterative negotiation, over multiple issues and agents, is modelled by the PERSUADER system through the concepts of argumentation and mediation (Sycara 1989). However, negotiation, as defined in this paper, is a *mutual* selection of outcome and precludes any intervention by outside parties. Furthermore, persuasion mechanisms operate on the beliefs of agents with the aim of changing one or both parties' beliefs. This is not the case for negotiation; it is not necessary for the agents to have similar beliefs at the end of negotiation.

Other systems such as KASBAH, (Chavez & Maes 1996), have attempted to engineer a real world application. KASBAH models time, actions and strategies involved in negotiation. However, negotiation in KASBAH is over a single issue and agents are semi-autonomous —the system models only a subset of the decision making that is involved in negotiation and the user ultimately makes the final decisions. Furthermore, the decisions that are delegated to the agents (called strategies in KASBAH) are limited to only three (c.f. the developed concessionary mechanism contains an infinite number of strategies) and their selection is not autonomous. Our model handles multiple issues and is designed for fully autonomous agents.

Conclusions

This paper has presented a distributed negotiation model that coordinates both agent interactions and individual agent decisions. Protocols have been defined that structure interactions and model the iterated nature of reaching agreements. Finally, mechanisms have been proposed for finding solutions which are based on realistic assumptions, are practical and model the complex nature of negotiation. The direction for future research is focused at empirical evaluation of the developed model to determine its applicability and performance profile.

References

Binmore, K. 1985. Bargaining and coalitions. In A.E.Roth., ed., *Game-Theoretic Models of Bargaining.* Cambridge University Press.

Chavez, A., and Maes, P. 1996. Kasbah: An agent marketplace for buying and selling goods. *Conference on Practical Applications of Intelligent Agents and Multi-Agent Technology.*

Davis, R., and Smith, R. 1988. Negotiation as a metaphor for distributed problem solving. In Bond, A., and Gasser, L., eds., *Readings in Distributed Artificial Intelligence.* Morgan Kaufmann Publishers, Inc. 333–357.

Faratin, P.; Sierra, C.; and Jennings, N. 1998. Negotiation decision functions for autonomous agents. *Robotics andAutonomous Systems* 24(3–4):159–182.

FIPA97. 1997. http://www.fipa.org/spec/FIPA97.html.

Pruitt, D. G. 1981. *Negotiation Behavior.* Academic Press.

Raiffa, H. 1982. *The Art and Science of Negotiation.* Cambridge, USA: Harvard University Press.

Sierra, C.; Faratin, P.; and Jennings, N. 1997. A service-oriented negotiation model between autonomous agents. In Boman, M., and de Velde, W. V., eds., *Multi-Agent Rationality: Proceedings of the 8th European Workshop on Modelling Autonomous Agents in Multi-Agent World, MAAMAW'97,* number 1237 in Lecture Notes in Artificial Intelligence, 17–35. Springer-Verlag.

Sycara, K. 1989. Multi-agent compromise via negotiation. In Gasser, L., and Huhns, M., eds., *Distributed Artificial Intelligence Volume II,* 119–139. San Mateo, California: Morgan Kaufmann.

Valverde, L. 1985. On the structure of F-indistinguishability. *Fuzzy Sets and Systems* 17:313–328.