Negotiation Decision Functions for Autonomous Agents

Peyman Faratin, Carles Sierra [1], Nick R. Jennings

Dept. Electronic Engineering, Queen Mary and Westfield College,
University of London, London E1 4NS, UK.
{P.Faratin,C.A.Sierra,N.R.Jennings}@qmw.ac.uk

Abstract

We present a formal model of negotiation between autonomous agents. The purpose of the negotiation is to reach an agreement about the provision of a service by one agent for another. The model defines a range of strategies and tactics that agents can employ to generate initial offers, evaluate proposals and offer counter proposals. The model is based on computationally tractable assumptions, demonstrated in the domain of business process management and empirically evaluated.

Keywords: Multi-agent systems, Negotiation, Business Process Management

1 Introduction

Autonomous agents are being increasingly used in a wide range of industrial and commercial domains [2]. These agents have a high degree of self determination – they decide for themselves what, when and under what conditions their actions should be performed. In most cases, such agents need to interact with other autonomous agents to achieve their objectives (either because they do not have sufficient capabilities or resources to complete their problem solving alone or because there are interdependencies between the agents). The objectives of these interactions are to make other agents undertake a particular course of action (e.g. perform a particular service), modify a planned course of action (e.g. delay or bring forward a particular action so that there is no longer a conflict), or come to an agreement on a common course of action. Since the

[1] On sabbatical leave from Artificial Intelligence Research Institute—IIIA, Spanish Council for Scientific Research—CSIC. 08193 Bellaterra, Barcelona, Spain. sierra@iiia.csic.es. With the support of the Spanish Ministry of Education grant PR95-313.

agents have no direct control over one another, they must persuade their acquaintances to act in particular ways (they cannot simply instruct them). The paradigm case of persuasion is negotiation – *a process by which a joint decision is made by two or more parties. The parties first verbalise contradictory demands and then move towards agreement by a process of concession making or search for new alternatives*, (cf. [9]).

Given its pervasive nature, negotiation comes in many shapes and forms. However in this work we are interested in a particular class of negotiation – namely *service-oriented negotiation*. In this context, one agent (the client) requires a service to be performed on its behalf by some other agent (the server) [2]. Negotiation involves determining a contract under certain terms and conditions. The negotiation may be iterative in that several rounds of offers and counter offers will occur before an agreement is reached or the negotiation is terminated.

When building an autonomous agent which is capable of flexible and sophisticated negotiation, three broad areas need to be considered [8] – what negotiation protocol will be used?, what are the issues over which negotiation takes place?, and what reasoning model will the agents employ? This paper concentrates predominantly on the final point although the protocol and negotiation object are briefly defined. A comprehensive reasoning model for service-oriented negotiation should determine: which potential servers should be contacted, whether negotiation should proceed in parallel with all servers or whether it should run sequentially, what initial offers should be sent out, what is the range of acceptable agreements, what counter offers should be generated, when negotiation should be abandoned, and when an agreement is reached.

To this end, this paper presents a formal account of a negotiating agent's reasoning component –in particular it concentrates on the processes of generating an initial offer, of evaluating incoming proposals, and of generating counter proposals. The model specifies the key structures and processes involved in this endeavour and defines their inter-relationships. The model was shaped by practical considerations and insights emanating from the development of a system of negotiating agents for business process management (see [6] and Section 2 for more details). The main contributions of this work are: (i) it allows rich and flexible negotiation schemes to be defined; (ii) it is based on assumptions which are realistic for autonomous computational agents (see Section 3.3 for the set of requirements and Section 7 for a discussion of related approaches), iii) it has been empirically evaluated (see Section 6) and (iv) it

[2] A service is a problem solving activity which has clearly defined start and end points. Examples include diagnosing a fault, buying a group of shares in the stock market, or allocating bandwidth to transmit a video-conference.

presents some initial results on the convergence of negotiation (this aspect is not discussed in this paper, but refer to [13] for more details).

In this paper we concentrate on many-parties, many-issues, single-encounter negotiations with an environment of limited resources (time among them). Section 2 gives details of the type of applications and scenarios we are interested in. Sections 3 to 5 present the proposed model and in section 6 the model is empirically evaluated. Finally, related work and some future avenues of work are outlined in sections 7 and 8 respectively.

2 Service-Oriented Negotiation

This section characterises a context in which service oriented negotiation takes place. The scenario is motivated by work in the ADEPT project [6] which has developed negotiating agents for business process management applications. However, we believe that the characteristics emerging from this domain have a wide variety of application. To provide a detailed context for this work, a multi-agent system for managing a British Telecom (BT) business process is presented (section 2.1). This scenario is then analysed in terms of its key characteristics and assumptions as they relate to the process of negotiation (section 2.2).

2.1 BT's Provide Customer Quote Business Process

This scenario is based on BT's business process of providing a quotation for designing a network to provide particular services to a customer (figure 1)[3]. The overall process receives a customer service request as its input and generates as its output a quote specifying how much it would cost to build a network to realise that service. It involves up to six agent types: the sales department agent, the customer service division agent, the legal department agent, the design division agent, the surveyor department agent, and the various agents who provide the out-sourced service of vetting customers.

The process is initiated by the sales agent which negotiates with the CSD agent (mainly over time, but also over the number of invocations and the form in which the final result should be delivered) for the service of providing a customer quote. The first stages of the $Provide\_Customer\_Quote$ service involve the CSD agent capturing the customer's details and vetting the customer in

[3] The negotiations between the agents are denoted by arrows (arrow head toward client) and the service involved in the negotiation is juxtaposed to the respective arrow.

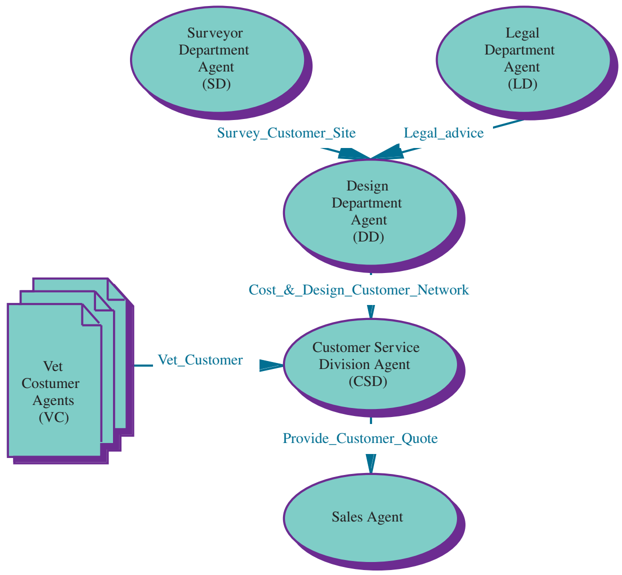

Fig. 1. Agent system for BT's provide customer quote business process

terms of their credit worthiness. The latter sub-service is actually performed by one of the VC agents. Negotiation is used to determine which VC agent should be selected – the main attributes negotiated over are the price of the service, the penalty for contract violation, the desired quality of the service and the time by which the service should be performed. If the customer fails the vetting procedure, then the quote process terminates. Assuming the customer is satisfactory, the CSD agent maps their requirements against a service portfolio. If the requirements can be met by a standard off-the-shelf portfolio item then an immediate quote can be offered based on previous examples. In the case of bespoke services, however, the process is more complex. The CSD agent negotiates with the DD agent (over time and quality) for the service of designing and costing the desired network service. In order for the DD agent to provide this service it must negotiate with the LD agent (over time) and perhaps with the SD agent. The LD agent checks the design to ensure the legality of the proposed service (e.g. it is illegal to send unauthorised encrypted messages across France). If the desired service is illegal, then the entire quote process terminates and the customer is informed. If the requested service is legal then the design phase can start. To prepare a network design it is usually necessary to have a detailed plan of the existing equipment at the customer's premises. Sometimes such plans might not exist and sometimes they may be out of date. In either case, the DD agent determines whether the customer site(s) should be surveyed. If such a survey is warranted, the DD agent negotiates with the SD agent (over price and time) for the $Survey\_Customer\_Site$ service. On completion of the network design and costing, the DD agent informs the CSD agent which informs the customer of the service quote. The business process then terminates.

The structure of the negotiation object is based almost directly on the legal contracts used to regulate agreements in the current manual approach to business process management. This structure is fairly rich and covers both service and meta-service attributes. In more detail, it contains: (i) the service name; (ii) a unique agreement identifier (covering the case where there are multiple agreements for the same service); (iii) the agents involved in the agreement (client and server); (iv) the type of agreement (one off agreement for a single service invocation versus on-going agreements for multiple invocations of the same service); (v) timing information (duration represents the maximum time the server can take to finish the service, and start time and end time represent the time during which the agreement is valid); (vi) the volume of invocations permissible between the start and end times (for on-going agreements only); (vii) the price paid per invocation; (viii) the penalty the server incurs for every violation of the agreement; (ix) the information the client must provide to the server on service invocation; and (x) the policy used for disseminating the service's intermediate and final results to the client.

2.2 Characteristics and Assumptions

The following negotiation characteristics can be noted from the ADEPT business process scenario. Moreover, it is believed that these characteristics are likely to be common to a wide range of service oriented negotiations between autonomous agents.

- A given service can be provided by more than one agent (e.g. multiple agents can provide the vet customer service to the CSD agent). The available services may be identical in their characteristics or they may vary along several dimensions (e.g. quality, price, availability, etc.).
- Individual agents can be both clients and servers for different services in different negotiation contexts.
- Negotiations can range over a number of quantitative (e.g. price, duration, and cost) and qualitative (e.g. type of reporting policy, and nature of the contract) issues. Each successful negotiation requires a range of such issues to be resolved to the satisfaction of both parties. Agents may be required to make trade-offs between issues (e.g. faster completion time for lower quality) in order to come to an agreement.
- The social context and inter-relationships of the participants influences the way agents negotiate. Some negotiations involve entities within the same organisation (e.g. between the CSD and DD agents) and hence are generally cooperative in nature. Other negotiations are inter-organisational and purely competitive – involving self interested, utility maximising agents (e.g. between the VC agents and the CSD agent). Some groups of agents often negotiate with one another for the same service (e.g. the CSD and DD agents),

whereas other negotiations are more open in nature (for example, the set of VC agents changes frequently and hence the CSD agent often negotiates with unknown agents).
- As the agents are autonomous, the factors which influence their negotiation stance and behaviour are private and not available to their opponents (especially in inter-organisational settings). Thus agents do not know what utilities their opponents place on various outcomes, they do not know what reasoning models they employ, they do not know their opponent's constraints and they do not know whether an agreement is even possible at the outset (i.e. the participants may have non-intersecting ranges of acceptability).
- Since negotiation takes place within a highly intertwined web of activity (the business process) time is a critical factor. Timings are important on two distinct levels: (i) the time it takes to reach an agreement must be reasonable; and (ii) the time by which the negotiated service must be executed is important in most cases and crucial in others. The former means that the agents should not become involved in unnecessarily complex and time consuming negotiations – the time spent negotiating should be reasonable with respect to the value of the service agreement. The latter means that the agents sometimes have hard deadlines by which agreements must be in place (this occurs mainly when multiple services need to be combined or closely coordinated).

3 The Negotiation Model

The negotiation model in this section is based on a variation of the *two parties, many issues* value scoring system presented in [10]. That is, a model for bilateral negotiations about a set of quantitative variables. Our variation transforms that model into a *many parties, many issues* model (that is, multilateral negotiations about a set of variables). This is important since multilateral negotiations are common in the application domains in which we are interested. Our model of multilateral negotiations is based on a set of mutually influencing *two parties, many issues* negotiations. We will call the sequence of offers and counter-offers in a two-party negotiation a *negotiation thread*. Offers and counter offers are generated by lineal combinations of simple functions, called *tactics*. Tactics generate an offer, or counter offer, for a single component of the negotiation object using a single criteria (time, resources, etc.). Different weights in the lineal combination allow the varying importance of the criteria to be modelled. For example, when determining values of slots in the negotiation object it may initially be more important to take into account the other agent's behaviour than the remaining time. In which case, the tactics that emphasize the behaviour of other agents will be given greater preference than the tactics which base their value on the amount of time remaining.

However, to achieve flexibility in the negotiation, the agents may wish to change their ratings of the importance of the different criteria over time. For example, remaining time may become correspondingly more important than the imitation of the other's behaviour as the time by which an agreement must be in place approaches. We use the term *strategy* to denote the way in which an agent changes the weights of the different tactics over time. Thus strategies combine tactics depending on the history of negotiations and the internal reasoning model of the agents, and negotiation threads influence one another by means of strategies (see Section 5).

Before presenting our model, we introduce Raiffa's basic model for bilateral negotiation [10] since this forms the basis of our work.

3.1 The bilateral negotiation model

Let i ($i \in \{a, b\}$) represent the negotiating agents and j ($j \in \{1, ..., n\}$) the issues under negotiation. Let $x_j \in [min_j, max_j]$ be a value for issue j. Here we limit ourselves to considering issues for which negotiation amounts to determining a value between a delimited range. Each agent has a scoring function $V_j^i : [min_j, max_j] \rightarrow [0, 1]$ that gives the score agent i assigns to a value of issue j in the range of its acceptable values. For convenience, scores are kept in the interval $[0, 1]$.

The next element of the model is the relative importance that an agent assigns to each issue under negotiation. w_j^i is the importance of issue j for agent i. We assume the weights of both agents are normalized, i.e. $\sum_{1 \leq j \leq n} w_j^i = 1$, for all i in $\{a, b\}$. With these elements in place, it is now possible to define an agent's scoring function [4] for a *contract* – that is, for a value $x = (x_1, ..., x_n)$ in the multi-dimensional space defined by the issues' value ranges:

$$V^i(x) = \sum_{1 \leq j \leq n} w_j^i V_j^i(x_j)$$

If both negotiators use such an additive scoring function, Raiffa showed it is possible to compute the optimum value of x as an element on the efficient frontier of negotiation [5] (see [10], p. 164).

[4] Non-linear approaches to modelling utility could be used if necessary without affecting the basic ideas of the model.

[5] Any contract not on this frontier is sub-optimal (i.e. not Pareto-optimal) in that possible mutual gains are missed.

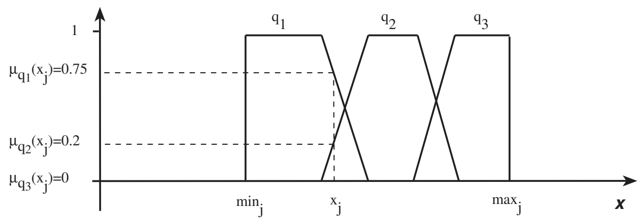

Fig. 2. Example of qualitative values representation with fuzzy sets

3.2 Qualitative values negotiation

As mentioned in section 2.1, some issues in negotiation such as the quality of a service are qualitative in nature rather than quantitative. We therefore designed two potential methods for incorporating qualitative issues into model.

3.2.1 Qualitative values

In this case, we define the scoring function directly over the finite set of qualitative values $Q = \{q_1, q_2, \ldots q_r\}$ as: $\overline{V_j} : Q \rightarrow [0, 1]$, a discrete function over the domain of qualities. This approach to representing qualities is merely a simplification of the general continuous case outlined in section 3.1.

3.2.2 Fuzzy values

Using this approach, we allow the issues to range over a continuous domain, as in the quantitative model (i.e $x_j \in [min_j, max_j]$), but the scoring function is defined as an interpolation between the membership functions of the fuzzy sets associated to the qualities Q [15]. That is, $q_i \in Q$ is a fuzzy set with a membership function μ_{q_i}, and

$$V_j(x_j) = \sum_{q_i \in Q} \mu_{q_i}(x_j)\overline{V_j}(q_i)$$

Thus we still require a function $\overline{V_j} : Q \rightarrow [0, 1]$ which establishes the relative importance of each one of the fuzzy values. However, we use the membership function value for x_j for each of the fuzzy sets in Q as a multiplying factor in order to obtain it's score value.

By way of an illustration consider the case depicted in figure 2. Assume $\overline{V_j}(q_1) = 0.2$, $\overline{V_j}(q_2) = 0.5$, and $\overline{V_j}(q_3) = 0.8$. We then have $V_j(x_j) = \mu_{q_1}(x_j)\overline{V_j}(q_1) + \mu_{q_2}(x_j)\overline{V_j}(q_2) + \mu_{q_3}(x_j)\overline{V_j}(q_3)$, that is $V_j(x_j) = 0.75 \times 0.2 + 0.2 \times 0.5 + 0 \times 0.8 = 0.25$. The main advantage of this approach over the pure qualitative method is the smooth way in which the scoring varies as the value of x_j changes.

3.3 Service-oriented negotiation requirements

The above bilateral negotiation model, suitably extended for qualitative values, maybe valid for some service oriented settings. However, the model contains several implicit assumptions that, although they permit good optimisation results, are inappropriate for our needs:

(i) *Privacy of information.* To find the optimum value, the scoring functions have to be disclosed. This is, in general, inappropriate for competitive negotiation.
(ii) *Privacy of models.* Both negotiators have to use the same additive scoring model. However, the models used to evaluate offers and generate counter offers are one of the things that negotiators try to hide from one another.
(iii) *Value restrictions.* There are pre-defined value regions for discussion (they are necessary to define the limits of the scoring function). However, it is not always possible to find these common regions and in many cases negotiation actually involves determining whether such regions even exist.
(iv) *Time restrictions.* There is no notion of timing issues in the negotiation. However, time is a major constraint on the agent's behaviour [7]. This is mainly true on the client side; agents often have strict deadlines by when the negotiation must be completed. For instance, a video link has to be provided at 16:00 because at that time a conference should start; negotiation about set up cannot continue after that time.
(v) *Resource restrictions.* There is no notion of resource issues in the negotiation. However, the quantity of a particular resource has a strong and direct influence on the behaviour of agents, and, moreover, the correct appreciation of the remaining resources is an essential characteristic of good negotiators. Resources from the client's point of view relate directly to the number of servers engaged in the ongoing negotiation; likewise from the server's point of view. Thus, the quantity of resource has a similar effect on the agents' behaviour as time.

Even just taking the first consideration alone, it is clear that optimal solutions cannot be found in our domains: it is not possible to optimize an unknown function. Hence, we shall propose a model for individual agent negotiation that seeks to find deals acceptable to its acquaintances but which, nevertheless, maximise the agent's own scoring function.

3.4 A service-oriented negotiation model

In service oriented negotiations, agents can undertake two possible roles that are, in principle, in conflict. Hence we shall distinguish (for notational convenience) two subsets of agents[6], $Agents = Clients \cup Servers$. We use roman letters to represent agents; $c, c_1, c_2, \ldots$ will stand for clients, $s, s_1, s_2, \ldots$ for servers and $a, a_1, b, d, e, \ldots$ for unspecific agents.

We adhere to an additive scoring system (section 3.1) in which, for simplicity, the function V_j^a is either monotonically increasing or monotonically decreasing.

In general, clients and servers have opposing interests, e.g. a client wants a low price for a service, whereas his potential servers attempt to obtain the highest price. High quality is desired by clients but not by servers, and so on. Therefore, in the space of negotiation values, negotiators represent opposing forces in each one of the dimensions. In consequence, the scoring functions verify that given a client c and a server s negotiating values for issue j, then if $x_j, y_j \in [min_j, max_j]$ and $x_j \geq y_j$ then $(V_j^c(x_j) \geq V_j^c(y_j)$ iff $V_j^s(x_j) \leq V_j^s(y_j))$. However, in a small number of cases the clients and service providers may have a mutual interest for a negotiation issue. For example, Raiffa cites a case [10, pg. 133–147] in which the Police Officers Union and the City Hall realize, in the course of their negotiations, that they both want the police commissioner fired. Having recognised this mutual interest they quickly agree that this course of action should be selected. Thus, in general, where there is a mutual interest, the variable will be assigned one of its extreme values. Hence these variables can be removed from the negotiation set. For instance, the act of firing the police commissioner can be removed from the set of issues under negotiation and assigned the extreme value "done".

Once the agents have determined the set of variables over which they will negotiate, the negotiation process between two agents ($a, b \in Agents$) consists of an alternate succession of offers and counter offers of values for these variables. This continues until an offer or counter offer is accepted by the other side or one of the partners terminates negotiation (e.g. because the time deadline is reached without an agreement being in place). Negotiation can be initiated by clients or servers.

We represent by $x^t_{a \rightarrow b}$ the vector of values proposed by agent a to agent b at time t, and by $x^t_{a \rightarrow b}[j]$ the value for issue j proposed from a to b at time t. The range of values acceptable to agent a for issue j will be represented as the interval $[min_j^a, max_j^a]$. For convenience, we assume a common global time (the calendar time) represented by a linearly ordered set of instants, namely $Time$, and a reliable communication medium introducing no delays in message transmission (so we can assume that transmission and reception times are identical). The common time assumption is not too strong for our application domains, because time granularity and offer and counter offers frequencies are not high. Then,

[6] The subsets are not disjoint since an agent can participate as a client in one negotiation and as a service provider in another.

Definition 1 *A* **Negotiation Thread** *between agents* $a, b \in Agents$*, at time* $t \in Time$*, noted* $x^t_{a \leftrightarrow b}$ *or* $x^t_{b \leftrightarrow a}$*, is any finite sequence of the form* $\{x^{t_1}_{d_1 \to e_1}, x^{t_2}_{d_2 \to e_2}, \ldots, x^{t_n}_{d_n \to e_n}\}$ *where:*

(i) $e_i = d_{i+1}$*, proposals are alternate between both agents,*
(ii) $t_k \leq t_l$ *if* $k \leq l$*, ordered over time,*
(iii) $d_i, e_i \in \{a, b\}$*, the thread contains only proposals between agents* a *and* b*,*
(iv) $d_i \neq e_i$*, the proposals are* between *agents, and*
(v) $x^{t_i}_{d_i \to e_i}[j] \in [min^{d_i}_j, max^{d_i}_j]$ *or is one of the particles* $\{accept, reject\}$*.*

Super-index t_n *represents an instant in the set* $Time$ *such that* $t_n \leq t$*. We will say that a negotiation thread is* **active** [7] *if* $x^{t_n}_{d_n \to e_n} \notin \{accept, reject\}$*.*

For simplicity in the notation, we assume that t_1 corresponds to the initial time value, that is $t_1 = 0$. In other words, there is a local time for each negotiation thread, that starts with the utterance of the first offer. When agent a receives an offer from agent b at time t, that is $x^t_{b \to a}$, it has to rate the offer using its scoring function. If the value of $V^a(x^t_{b \to a})$ is greater than the value of the counter offer agent a is ready to send at the time t' when the evaluation is performed, that is, $x^{t'}_{a \to b}$ with $t' > t$, then agent a accepts. Otherwise, the counter offer is submitted. The interpretation function I^a expresses this concept more formally:

Definition 2 *Given an agent* a *and its associated scoring function* V^a*, the* **interpretation** *by agent* a *at time* t' *of an offer* $x^t_{b \to a}$ *sent at time* $t < t'$*, is defined as:*

$$I^a(t', x^t_{b \to a}) = \begin{cases} reject \;\; \mathit{If}\; t' > t^a_{max} \\ accept \;\; \mathit{If}\; V^a(x^t_{b \to a}) \geq V^a(x^{t'}_{a \to b}) \\ x^{t'}_{a \to b} \quad otherwise \end{cases}$$

where $x^{t'}_{a \to b}$ *is the contract that agent* a *would offer to* b *at the time of the interpretation, and* t^a_{max} *is a constant that represents the time by which* a *must have completed the negotiation.*

The result of $I^a(t', x^t_{b \to a})$ is used to extend the current negotiation thread between the agents. This interpretation formulation also allows us to model the fact that a contract unacceptable today can be accepted tomorrow merely by the fact that time has passed.

[7] We assume that any offer is valid (that is, the agent that uttered it is commited) until a counter offer is received. If the response time is relevant it can be included in the set of issues under negotiation.

In order to prepare a counter offer, $x^{t'}_{a \to b}$, agent a uses a set of tactics that generate new values for each variable in the negotiation set. Based on the needs of our business process applications (Section 2), we developed the following families of tactics:

(i) **Time-dependent**. If an agent has a time deadline by which an agreement must be in place, these tactics model the fact that the agent is likely to concede more rapidly as the deadline approaches. The shape of the curve of concession, a function depending on time, is what differentiates tactics in this set.
(ii) **Resource-dependent**. These tactics model the pressure in reaching an agreement that the limited resources –e.g. remaining bandwidth to be allocated, money, or any other– and the environment – e.g number of clients, number of servers or economic parameters– impose upon the agent's behaviour. The functions in this set are similar to the time dependent functions except that the domain of the function is the quantity of resources available instead of the remaining time.
(iii) **Imitative**. In situations in which the agent is not under a great deal of pressure to reach an agreement, it may choose to use imitative tactics that protect it from being exploited by other agents. In this case, the counter offer depends on the behaviour of the negotiation opponent. The tactics in this family differ in which aspect of their opponent's behaviour they imitate, and to what degree the opponent's behaviour is imitated.

We do not claim that these family types are complete, nor that we have enumerated all possible instances of tactics within a given family. Rather these are merely the types of tactics we found useful in our applications.

4 Negotiation tactics

Tactics are the set of functions that determine how to compute the value of an issue (price, volume, duration, quality, ...), by considering a *single* criteria (time, resources, ...). The set of values for the negotiation issue are then the range of the function, and the single criteria is its domain. The criteria we have chosen, as explained in the previous section, are time, resources and previous offers and counter offers.

Given that agents may want to consider more than one criterion to compute the value for a single issue, we model the generation of counter proposals as a weighted combination of different tactics covering the set of criteria. The values so computed for the different issues will be the elements of the counter proposal[8]. For instance, if an agent wants to counter propose taking into account two criteria: the remaining time and the previous behaviour of the opponent, it can select two tactics: one from the time-dependent family and one from the imitative family. Both of these tactics will suggest a value to counter propose for the issue under negotiation. The actual value which is counter proposed will be the weighted combination of the two suggested values.

To illustrate these points consider the following example. Given an issue j, for which a value is under negotiation, an agent a's initial offer corresponds to a value in the issue's acceptable region, (i.e a value in $[min_j^a, max_j^a]$). For instance, if $a's$ range is [*£0*, *£20*] for the price p to pay for a good then it may start the negotiation process by offering the server *£10* –what initial offer should be chosen is something the agent can learn by experience. The server, agent b, with range [*£17*, *£35*] may then make an initial counter-offer of *£25*. With these two initial values, the strategy of agent a may consist of using a (single criterion) time-dependent tactic which might make a reasonably large concession and suggest *£15* since it does not have much time in which to reach an agreement. Agent b, on the other hand, may be using two criteria to compute it's counterproposal – e.g a time-dependent tactic (which might suggest a small concession to *£24* since it has a long time until the deadline) and an imitative tactic (which might suggest a value of *£20* to mirror the *£5* shift of the opponent). If agent b rates the time-dependent behaviour three times as important as the imitative behaviour, then the value of the counter-offer will be $(0.75 * 24) + (0.25 * 20) =$ *£23*. This process continues until the agents converge on a mutually acceptable solution.

It should be noted that not all tactics can be applied at all instants. For instance, a tactic that imitates the behaviour of an opponent is only applicable when the opponent has shown its behaviour sufficiently. For this reason the following description of the tactics pays particular attention to their applicability conditions.

4.1 Time-dependent tactics

In these tactics, the predominant factor used to decide which value to offer next is time, t. Thus these tactics consist of varying the acceptance value for the issue depending on the remaining negotiation time (an important requirement in our domain —Section 2.2), modelled as the above defined constant t^a_{max}. We model the initial offer as being a point in the interval of values of the issue under negotiation. Hence, agents define a constant κ_j^a that multiplied by the

[8] Values for different issues may be computed by different weighted combinations of tactics.

size of the interval determines the value of issue j to be offered in the first proposal by agent a.

We model the value to be uttered by agent a to agent b for issue j as the offer at time t, with $0 \leq t \leq t^a_{max}$, by a function α^a_j depending on time as the following expression shows:

$$x^t_{a \rightarrow b}[j] = \begin{cases} min^a_j + \alpha^a_j(t)(max^a_j - min^a_j) & \text{If } V^a_j \text{ is decreasing} \\ min^a_j + (1 - \alpha^a_j(t))(max^a_j - min^a_j) & \text{If } V^a_j \text{ is increasing} \end{cases}$$

A wide range of time-dependent functions can be defined simply by varying the way in which $\alpha^a_j(t)$ is computed. However, functions must ensure that $0 \leq \alpha^a_j(t) \leq 1$, $\alpha^a_j(0) = \kappa^a_j$ and $\alpha^a_j(t^a_{max}) = 1$. That is, the offer will always be between the value range, at the beginning it will give the initial constant and when the time deadline is reached the tactic will suggest to offer the reservation value [9] . We distinguish two families of functions with this intended behaviour: polynomial and exponential (naturally, others could also be defined). Both families are parameterised by a value $\beta \in \Re^+$ that determines the convexity degree (see Figure 3) of the curve. We chose these two families of functions because of the very different way they model concession. For the same big value of β, the polynomial function concedes faster at the beginning than the exponential one, then they behave similarly. For a small value of β, the exponential function waits longer than the polynomial one before it starts conceding

- **Polynomial.** $\alpha^a_j(t) = \kappa^a_j + (1 - \kappa^a_j)(\frac{min(t,t_{max})}{t_{max}})^{\frac{1}{\beta}}$
- **Exponential.** $\alpha^a_j(t) = e^{(1-\frac{min(t,t_{max})}{t_{max}})^{\beta} \ln \kappa^a_j}$

These families of functions represent an infinite number of possible tactics, one for each value of β. However to better understand their behaviour we have classified them, depending on the value of β, into two extreme sets showing clearly different patterns of behaviour. Other sets in between these two could also be defined:

[9] The reservation value for issue j of agent a represents the value that gives the smallest score for function V^a_j. The reservation value for agent a and issue j depends on the function V^a_j and the range $[min^a_j, max^a_j]$. If V^a_j is monotonically increasing, then the reservation value is min^a_j; if it is decreasing the reservation value is max^a_j.

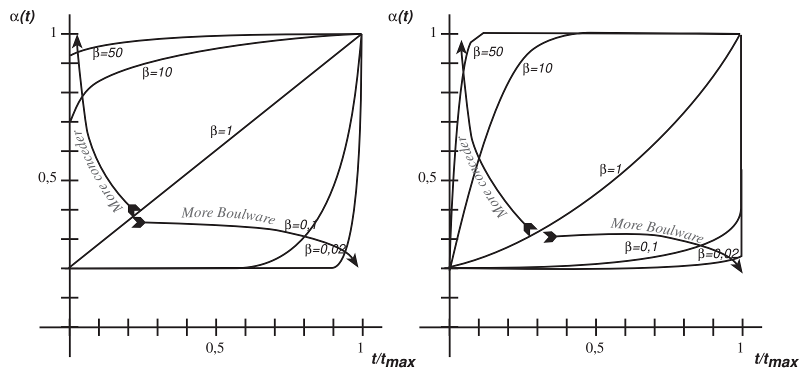

Fig. 3. Polynomial (left) and Exponential (right) functions for the computation of $\alpha(t)$. Time is presented as relative to t^a_{max}.

(i) **Boulware tactics [[10], pg. 48]**. Either exponential or polynomial functions with $\beta < 1$. This tactic maintains the offered value until the time is almost exhausted, whereupon it concedes up to the reservation value[10]. The behaviour of this family of tactics with respect to β is easily understood taking into account that $\lim_{\beta \to 0^+} e^{(1-\frac{min(t,t_{max})}{t_{max}})^\beta \ln \kappa^a_j} = \kappa^a_j$ or $\lim_{\beta \to 0^+} \kappa^a_j + (1-\kappa^a_j)(\frac{min(t,t_{max})}{t_{max}})^{\frac{1}{\beta}} = \kappa^a_j$.

(ii) **Conceder [[9], pg. 20]**. Either exponential or polynomial functions with $\beta > 1$. The agent goes to its reservation value very quickly. For similar reasons as before, we have $\lim_{\beta \to +\infty} e^{(1-\frac{min(t,t_{max})}{t_{max}})^\beta \ln \kappa^a_j} = 1$ or $\lim_{\beta \to +\infty} \kappa^a_j + (1-\kappa^a_j)(\frac{min(t,t_{max})}{t_{max}})^{\frac{1}{\beta}} = 1$.

4.2 Resource-dependent tactics

These tactics are similar to the time-dependent ones. Indeed time-dependent tactics can be seen as a type of resource-dependent tactic in which the sole resource considered is time. Whereas time vanishes constantly up to its end, other resources may have different patterns of usage. We model resource-dependent tactics in the same way as time-dependent ones, that is, by using the same functions, but by either:

(i) making the value of t^a_{max} dynamic (section 4.2.1), or
(ii) making the function α depend on an estimation of the amount of a particular resource (section 4.2.2).

[10] Besides the pattern of concession that these functions model, Boulware negotiation tactics presume that the interval of values for negotiation is narrow. Hence, when the deadline is reached and $\alpha(t_{max}) = 1$ the offer generated is not substantially different from the initial one.

4.2.1 Dynamic-deadline tactics

The dynamic value of t^a_{max} represents a heuristic about how many resources are in the environment. The scarcer the resource, the more urgent the need for an agreement. In our application domains, the most important resource to model is the number of agents negotiating with a given agent and how keen they are to reach agreements. On one hand, the greater the number of agents who are negotiating with agent a for a particular service s, the lower the pressure on agent a to reach an agreement with any specific individual. While on the other hand, the longer the negotiation thread, the greater the pressure on a to come to an agreement. Hence, representing the set of agents negotiating with agent a at time t as: $N^a(t) = \{i | x^t_{i \leftrightarrow a} \text{is active}\}$, we define the dynamic time deadline for agent a as:

$$t^a_{max} = \mu^a \frac{|N^a(t_c)|^2}{\sum_i |x^{t_c}_{i \leftrightarrow a}|}$$

where μ^a represents the time agent a considers reasonable to negotiate with a single agent and $|x^{t_c}_{i \leftrightarrow a}|$ represents the length of the current thread between i and a. Notice that the number of agents is in the numerator – so quantity of time is directly proportional to it, and averaged length of negotiation thread is in the denominator – so quantity of time is inversely proportional to it.

4.2.2 Resource-estimation tactics

These tactics generate counter-offers depending on how a particular resource is being consumed. Resources could be money being transferred among agents, the number of agents interested in a particular negotiation, and also, in a similar way as before, time. We want the agent to become progressively more conciliatory as the quantity of resource diminishes. The limit when the quantity of the resource approaches nil is to concede up to the reservation value for the issue(s) under negotiation. When there is plenty of resource, a more Boulware behaviour is to be expected. Formally, this can be modelled by having a different computation for the function α:

$$\alpha^a_j(t) = \kappa^a_j + (1 - \kappa^a_j) e^{-resource(t)}$$

where the function $resource(t)$ measures the quantity of the resource at time t. Examples of functions are:

$$resource(t) = |N^a(t)|.$$
$$resource(t) = \mu^a \frac{|N^a(t)|^2}{\sum_i |x^t_{i \leftrightarrow a}|}.$$
$$resource(t) = \min(0, t_{max} - t)$$

In the first example, the number of negotiating agents is the resource. That is, the more agents negotiating the less pressure in making concessions. The second example models time as a resource in a similar way as in the previous section. The more agents, the less pressure, and the longer the negotiations the more pressure. Finally, the last case is modelling time again as a resource, but in this case the quantity of resource decreases lineally with respect to time.

4.3 Behaviour-dependent tactics

This family of tactics compute the next offer based on the previous attitude of the negotiation opponent. These tactics have proved important in co-operative problem-solving negotiation settings [1], and so are useful in a subset of our contexts (see Section 2.2). The main difference between the tactics in this family is in the type of imitation they perform. One family imitates proportionally, another in absolute terms, and the last one computes the average of the proportions in a number of previous offers. Hence, given a negotiation thread $\{\ldots, x^{t_{n-2\delta}}_{b\to a}, x^{t_{n-2\delta+1}}_{a\to b}, x^{t_{n-2\delta+2}}_{b\to a}, \ldots, x^{t_{n-2}}_{b\to a}, x^{t_{n-1}}_{a\to b}, x^{t_n}_{b\to a}\}$, with $\delta \geq 1$, we distinguish the following families of tactics:

(i) **Relative Tit-For-Tat**
The agent reproduces, in percentage terms, the behaviour that its opponent performed $\delta \geq 1$ steps ago. The condition of applicability of this tactic is $n > 2\delta$.

$$x^{t_{n+1}}_{a\to b}[j] = min(max(\frac{x^{t_{n-2\delta}}_{b\to a}[j]}{x^{t_{n-2\delta+2}}_{b\to a}[j]} x^{t_{n-1}}_{a\to b}[j], min^a_j), max^a_j)$$

(ii) **Random Absolute Tit-For-Tat**
The same as before but in absolute terms. It means that if the other agent decreases its offer by *£2*, then the next response should be increased by the same *£2*. Moreover, we add a component that modifies that behaviour by increasing or decreasing (depending on the value of parameter s) the value of the answer by a random amount. (This is introduced as it can enable the agents to escape from local minima.) M is the maximum amount by which an agent can change its imitative behaviour. The condition of applicability is again $n > 2\delta$.

$$x_{a\to b}^{t_{n+1}}[j] =$$
$$min(max(x_{a\to b}^{t_{n-1}}[j] + (x_{b\to a}^{t_{n-2\delta}}[j] - x_{b\to a}^{t_{n-2\delta+2}}[j]) + (-1)^s R(M), min_j^a), max_j^a)$$

where

$$s = \begin{cases} 0 \text{ If } V_j^a \text{ is decreasing} \\ 1 \text{ If } V_j^a \text{ is increasing} \end{cases}$$

and $R(M)$ is a function that generates a random integer in the interval $[0, M]$.

(iii) **Averaged Tit-For-Tat** The agent computes the average of percentages of changes in a window of size $\gamma \geq 1$ of its opponents history when determining its new offer. When $\gamma = 1$ we have the relative Tit-For-Tat tactic with $\delta = 1$. The condition of applicability for this tactic is $n > 2\gamma$.

$$x_{a\to b}^{t_{n+1}}[j] = min(max(\frac{x_{b\to a}^{t_{n-2\gamma}}[j]}{x_{b\to a}^{t_n}[j]} x_{a\to b}^{t_{n-1}}[j], min_j^a), max_j^a)$$

5 Negotiation strategies

The aim of agent a's negotiation strategy is to determine the best course of action which will result in an agreement on a contract x that maximises its scoring function V^a. In practical terms, this equates to how to prepare a new counter offer.

In our model we consider that the agent has a representation of its mental state containing information about its beliefs, its knowledge of the environment (time, resources, etc.), and any other attitudes (desires, goals, obligations, intentions, etc.) the agent designer considers appropriate [11]. The mental state of agent a at time t is noted as MS_a^t. We denote the set of all possible mental states for agent a as MS_a.

When agent a receives an offer from agent b it becomes the last element in the current negotiation thread between both agents. If the offer is unsatisfactory, agent a generates a counter offer. As discussed earlier, different combinations of tactics can be used to generate counter offers for particular issues. An agent's strategy determines which combination of tactics should be used at any one instant. Hence, the following definition:

[11] We do not prescribe a particular mental state but rather aim towards an architecturally neutral description.

Definition 3 *Given a negotiation thread between agents* a *and* b *at time* t_n *over domain* $X = X_1 \times \ldots \times X_p$*, with* $x^{t_n}_{a \leftrightarrow b} = \{\ldots, x^{t_n}_{b \rightarrow a}\}$*, and a finite set of* m *tactics*[12] $T^a = \{\tau_i | \tau_i : MS_a \rightarrow X\}_{i \in [1,m]}$*, a* **weighted counter proposal** *is any lineal combination of the tactics that generates the value at time* t_{n+1} *in the thread. That is, for issue* j

$$x^{t_{n+1}}_{a \rightarrow b}[j] = \gamma_{j1}\tau_1(MS^{t_{n+1}}_a)[j] + \gamma_{j2}\tau_2(MS^{t_{n+1}}_a)[j] + \ldots + \gamma_{jm}\tau_m(MS^{t_{n+1}}_a)[j]$$

such that for all issues j*,* $\sum_{i \in [1,m]} \gamma_{ji} = 1$ *and* $x^{t_{n+1}}_{a \leftrightarrow b} = \{\ldots, x^{t_n}_{b \rightarrow a}, x^{t_{n+1}}_{a \rightarrow b}\}$

Given a set of tactics, different types of negotiation behaviour can be obtained by weighting the tactics in a different way. That is, by changing the matrix Γ – particular to each negotiation thread:

$$\Gamma^t_{a \rightarrow b} = \begin{pmatrix} \gamma_{11} & \gamma_{12} & \cdots & \gamma_{1m} \\ \gamma_{21} & \gamma_{22} & \cdots & \gamma_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ \gamma_{p1} & \gamma_{p2} & \cdots & \gamma_{pm} \end{pmatrix}$$

An example of when this weighted combination may be useful is when modelling a *smooth* transition from a behaviour based on a single tactic (e.g. Boulware, because the agent has plenty ot time to reach an agreement) to another one (e.g. Conceder, because the time is running out). Smoothness is obtained by changing the weight affecting the tactics progressively (e.g. from 1 to 0 and from 0 to 1 in the example).

We model many-parties negotiations by means of a set of interacting negotiation threads. The way this is done is by making a negotiation thread influence the selection of which matrix Γ is to be used in other negotiation threads. Thus,

Definition 4 *Given* $a, b \in Agents$*,* $t \in Time$*,* a*'s mental state* MS^t_a*, and* $\Gamma^t_{a \rightarrow b}$*, a* **Negotiation Strategy***, is any function* f *of the following type:*

$$\Gamma^{t+1}_{a \rightarrow b} = f(\Gamma^t_{a \rightarrow b}, MS^t_a)$$

[12] This definition uses the natural extension of tactics to the multi-dimensional space of issues' values.

A simplistic example of the application of our model would be to have a matrix Γ built up of 0s and 1s and having $\Gamma^{t+1}_{a \rightarrow b} = \Gamma^{t}_{a \rightarrow b}$ for all t. This would correspond to using a fixed single tactic for each issue at every instant in the negotiation.

6 Experimental Evaluation of the Negotiation Model

The model we have presented defines and formalises a range of negotiation behaviours. However, we canot say from the theoretical model alone which of these behaviours will be successful in which negotiation contexts (since there are too many interrelated variables and too wide a range of situations to consider). Therefore our approach is to empirically evaluate the main parameters of the model with the final aim of determining the most successful behaviours in various types of situations. At this stage, however, our investigation is focused on determining the behaviour and inter-dependencies of the model's basic constituent elements. This analysis will then lay the foundation for subsequent experimental work. To this end, we concentrate solely on the behaviour of pure tactics (i.e we exclude strategies that combine several tactics).

The experiments involve selecting a particular tactic, generating a range of random environments, then allowing the agent to negotiate using the chosen tactic against an opponent who employs a range of other tactics. Various experimental measures related to the negotiations are then recorded. In particular, Section 6.1 defines the experimental environments and the tactics, Section 6.2 describes the experimental measures, and finally section 6.3 describes the experimental hypotheses, the procedures and discussion of the results.

6.1 Environments and Tactics

Negotiation takes place in a particular context or environment. Environments are characterised by the number of agents they contain, the issues which are being discussed, the deadlines by when agreements must be reached, and the expectations of the agents. Since there are infinitely many potential environments, we need to find a means of selecting a representative and finite subset in which we can assess an agent's negotiation performance.

To this end, the first simplification involves limiting ourselves to bilateral negotiation between a single client and server over the single issue of *price*. Given this situation, the experimental environment is uniquely defined by the following variables: $[t^c_{max}, t^s_{max}, \kappa^c, \kappa^s, min^c_{price}, max^c_{price}, min^s_{price}, max^s_{price}]$. We compute the negotiation interval (the difference between the agent's minimum and maximum values) for price using two variables: θ^a and Φ. The variable θ^a is the length of the negotiation interval for an agent a. Φ is the degree of intersection between the negotiation intervals of the two agents and it ranges between 0 (full overlap) and 0.99 (almost non-overlapping reservations). For each environment, we assigned $min^c_{price} = 10$ and randomly select θ^a between the ranges of $\{10, 30\}$ for both agents. We then compute the negotiation intervals in the following way:

$$max^c = min^c + \theta^c$$

$$min^s = \theta^c\Phi + min^c$$

$$max^s = min^s + \theta^s$$

Note that the server's minimum reservation value is never lower than the client's minimum. This is because we are not interested in degenerate negotiation contexts where a server's offer is below a client's minimum and is immediately accepted. This method of generating reservation values also means a deal is always possible since there is always some degree of overlap.

The second simplification involves selecting a finite range of tactics since the theoretical model allows for an infinite set (e.g the range of β is infinite which means there are infinitely many time-dependent tactics). Therefore for analytical tractability we let the set of tactics used in the experiments be bound to a limited number of groups whose members could, however, be equally selected for a game. We chose nine experimental tactic groups; three each from the time-dependent, resource-dependent and behaviour-dependent families. We chose an equal number for each family to ensure that the tactic population is uniform and the results are not skewed by having more encounters with a particular type of tactic.

The three time-dependent families were chosen with ranges for parameter β which correspond to a Boulware, a Linear and a Conceder (β was respectively sampled between $[0.01, 0.2]$, $[1, 1]$ and $[20, 40]$). We chose the exponential function for Boulware because at low values of β they are more Boulware than the polynomials. However, polynomial function were selected for Linears and Conceders because only the polynomial function produces a linear behaviour at $\beta = 1$ (see figure 3) and because to Conceders they are more conciliatory initially than the exponentials. The three resource-dependent tactics (Thread1, Thread2 and Thread3) view the length of the thread as their main resource. The differences between these tactics relate to the amount of time they consider reasonable for negotiation (set by the parameter μ). To compare the behaviours of this family we set $\mu = 1$ for Thread1 and sampled between the ranges $[6, 10]$ for Thread2 and between $[11, 15]$ for Thread3. Since we only have one client and one server the value of parameter N remains constant at 1. Finally, relative, random and average titfortat tactics represent the behaviour-dependent family. We chose $\delta = 1$ for the relative and random tactics and $\gamma = 2$ for the average tactics. We did not sample these tactics because we want them to be applicable at the earliest possible opportunity. When a behaviour-dependent tactic is not applicable, the policy is to offer the value suggested by κ^a as the first offer then concede by a certain amount [1]. We chose a polynomial time-dependent tactic to compute the next offer with $\beta \in \{1, 3\}$. For random tactics we let $R(M)$ be sampled between 1 to 3 percent of the initial offer.

Each tactic group is then sampled for *every* environment since we are interested in the behaviour of tactic families rather than single concrete tactics. For each environment e_k, where k indexes the various environments, we define two matrices representing the outcomes of the client, $game_c^{e_k}$, and the server, $game_s^{e_k}$, when playing particular tactics. We index the client's tactics by the rows i and the server's by the columns j, so $game_c^{e_k}[i, j]$ is the outcome of the client when playing tactic i with server playing tactic j. Each tactic plays against all other tactics in each environment, hence $1 \leq i, j \leq 9$.

6.2 Experimental Measures

To evaluate the effectiveness of the tactics we consider the following measures which calibrate: i) the intrinsic benefit of the tactic family to an agent (section 6.2.1); ii) the cost adjusted benefit which moderates the intrinsic benefit with some measure of the cost involved in achieving that benefit (section 6.2.2) and iii) the performance of the intrinsic utility relative to a game of perfect information (section 6.2.3).

6.2.1 Intrinsic Agent Utility

The intrinsic benefit is modeled as the agent's utility for the negotiation's final outcome when using a tactic family in a given environment independently of the time taken and the resources consumed [12]. This utility, $U_a^{e_k}$, is calculated for each agent for a price x as:

$$U_c^{e_k}(x) = min(1, (\frac{max^c_{price} - x}{max^c_{price} - min^c_{price}})^{\frac{1}{\beta}})$$
$$U_s^{e_k}(x) = min(1, (\frac{x - min^s_{price}}{max^s_{price} - min^s_{price}})^{\frac{1}{\beta}})$$

where β determines the degree of convexity of the utility function. In this case, we chose $\beta = 1$ corresponding to a linear scoring function for all agents [13].

[13] We chose to model utilities as polynomials so that we could experiment with non-linear utilities in future.

If no deal is made in a particular negotiation, then we assign zero to both $U_c^{e_k}$ and $U_s^{e_k}$. However, by defining agent utilities in this manner we could not distinguish between deals made at reservations and no deals. Therefore to evaluate a tactic's performance we also computed intrinsic utility for cases in which a deal was indeed made.

The outcomes of the negotiations, as presented in the previous subsection, are represented in the matrices $game_a^{e_k}$. Hence the utility for a client c when negotiating using a tactic i against a server s using tactic j in environment e_k is $U_c^{e_k}(game_c^{e_k}[i,j])$.

6.2.2 *Cost Adjusted Benefit*

In addition to knowing the intrinsic utility to an agent for using a particular tactic, we are also interested in knowing the relationship between an outcome's utility and the costs the tactic imposes on the agents. Therefore the cost adjusted benefit of tactic pairs i and j in environment e_k is defined as follows:

$$B_a^{e_k}[i,j] = U_a^{e_k}[i,j] - C_a^{e_k}[i,j]$$

To define the cost function, C, we introduce the notion of a *system*. A system in these experiments is a set of resources that can be used by the agents during their negotiations. The usage of these system resources is subject to a tax $\mathcal{T}$ which is levied on each message communicated between the agents. Therefore the greater the communication between the agents, the greater the cost to the agents, and the greater the system revenue. The amount of communication in a particular negotiation is simply the total number of messages exchanged between the agents. So, we define the cost to an agent as:

$$C_c^{e_k}[i,j] = C_s^{e_k}[i,j] = \tanh(|x_{c_i \leftrightarrow s_j}| * \mathcal{T})$$

where $|x_{c_i \leftrightarrow s_j}|$ is the length of the thread at the end of negotiation between a client using tactic i and and a server using tactic j, tanh is an increasing function that maps the real numbers into $[0,1]$. $\mathcal{T}$ determines the rate of change of tanh(). We sampled $\mathcal{T}$ between the ranges of $[0.001, 0.1]$. In short, the greater the taxation system, the more costly the communication, and the quicker the rate at which the cost rises to an agent for each message.

6.2.3 *Experimental Controls: The Perfect Information Game*

All the measures so far have been designed to calibrate the performance of the tactics within our model. However, we also need to calibrate the performance of our tactics with respect to some control conditions so that we can relate our work with that of others. Game Theory, Economics and Voting Theory have all proposed desirable properties and solution criteria that can be used to characterise an agent's negotiation. Typically, these properties and criteria are concerned with the influence of the individual agent on the outcome or conversely the influence of the outcome on the individual. Specific measures include: Pareto optimality, symmetry, fairness and individual rationality [4]. We chose to compare the outcome attained by a pair of our tactic families with the outcome suggested by a protocol in which the agents declare their true reservation prices at the first step of negotiation and then share the overlap in the declared reservation values. This choice is fair and Pareto optimal in that the outcome is beneficial to both agents and any deviation from this outcome results in an increase in utility to one agent at the cost of a decrease in utility to the other [3]. For example, consider a client agent c and a server agent s having price reservation values $[min^c_{price}, max^c_{price}]$ and $[min^s_{price}, max^s_{price}]$ respectively and $max^c_{price} \geq min^s_{price}$. We then define the control outcome $\mathcal{O}^{e_k}$ for a given environment e_k as:

$$\mathcal{O}^{e_k} = \frac{max^c_{price} + min^s_{price}}{2}$$

Applying the definitions of utility presented before we get:

$$U^{e_k}_c(\mathcal{O}^{e_k}) = \frac{max^c_{price} - \mathcal{O}^{e_k}}{max^c_{price} - min^c_{price}}$$

$$U^{e_k}_s(\mathcal{O}^{e_k}) = \frac{\mathcal{O}^{e_k} - min^s_{price}}{max^s_{price} - min^s_{price}}$$

We then define the comparative performance of agents using our model with respect to the one shot protocol as the difference between the intrinsic agent utility (6.2.1) and the utility the agent would have received in the control protocol:

$$Gain^{e_k}_a[i,j] = U^{e_k}_a(game^{e_k}_a[i,j]) - U^{e_k}_a(\mathcal{O}^{e_k})$$

6.2.4 Average Utilities

To produce statistically meaningful results, we analyse the influence of each tactic family, on the above utilities, averaged across a number of environments and summed against all other tactics for each agent. Therefore our analysis is based on the performance of a tactic family across all other tactic families. The precise set of environments is sampled from the parameters specified in section 6.1 and the number of environments used is 200 with the probability of the sample mean deviating by more than 0.01 from the true mean being less than 0.05.

The average utility for clients was updated at the end of every game played in each environment according to:

$$\bar{U}_c^{e_k}[i,j] = \bar{U}_c^{e_{k-1}}[i,j]\frac{k-1}{k} + \frac{U(game_c^{e_k}[i,j])}{k}$$

and similarly for servers. The other measures were also handled in the same way.

6.3 Hypotheses and Results

The experiments considered here relate to two main components of the negotiation model: i) the amount of time available to make an agreement, t^a_{max} and ii) the relative value of the initial offer, κ^a. To test the effects of varying deadlines on agreements, we classify the experiments into environments where the time to reach an agreement is large (6.3.1) and environments where the time is small (6.3.2). Likewise for initial offers. We classify the environments into two types: those where the initial offer is distributed at the minimum of the agent's reservation values and those where the initial offer is distributed near the maximum of the reservation values (6.3.3). The reservation values were computed as described in section 6.1 with $\theta^c = \theta^s = 30$ and $\Phi = 0$. The reader is referred to figure 4 for the key to the experimental tactics. Each key is further postfixed by the role of an agent (e.g BC and BS denote a client and a server playing a Boulware tactic respectively).

To help clarify the behaviour of the agents, we trace the behaviour of time-dependent and resource-dependent tactics for different deadlines in figure 5 [14]. Because the behaviour of imitative tactics is a function of their interactions with other tactics in the population, we do not include them in this figure. For a similar reason, most of the discussion of results will be based on the behaviour of the time and resource dependent tactics since it is their behaviour which determines the behaviours of imitative tactics.The gradient of approach to reservation values increases as time decreases for the time-dependent tactics making arrival to the cross over in acceptance levels (a point in time at which submitted bids are mutually acceptable) ealier in time. Boulware is almost a step function which begins to concede much later than the others. Linear and Conceder approach their reservation values at an even a faster rate when their deadlines are reduced. The behaviour of resource-dependent tactics, on the other hand, is independent of the time limit. All three tactics initially have an almost linear rate of change which begins to reach an asymptote as they approach their reservation values. They differ, from one another, in that none

[14] Because we cannot plot behaviours across sampled environments, we represent the behaviour profiles for a single environment.

Tactic Family	*Abbreviation*	*Experimental Parameters Ranges*
Boulware	B	$\beta \in \{0.01, 0.2\}$
Boulware	B2	$\beta \in \{0.25, 0.4\}$
Boulware	B3	$\beta \in \{0.4, 0.7\}$
Boulware	B4	$\beta \in \{0.7, 0.99\}$
Linear	L	$\beta = 1.0$
Conceder	C	$\beta \in \{20.0, 40.0\}$
Thread1	T1	$\mu = 1$, n $= 1$
Thread2	T2	$\mu \in \{1, 5\}$, n $= 1$
Thread3	T3	$\mu \in \{5, 10\}$, n $= 1$
Relative tit for tat	RE	$\delta = 1$
Random tit for tat	RA	$\delta = 1$ m $\in \{1, 3\}$
Average tit for tat	AV	$\gamma = 2$

Fig. 4. Experimental Tactic Key

of them offers their reservation values and as μ increases the final offers are distributed further away from the reservation values.

6.3.1 Long Term Deadlines

Our hypotheses about the effect of long term deadlines can be stated as follows:

Hypothesis1: *In environments where the deadline for negotiation is large, tactics which quickly approach their reservation values will make more deals than tactics which have a slower rate of approach. However, the deals made will be at the reservation values and so will have less intrinsic utility than those tactics whose rate of approach is slower.*

Hypothesis2: *The utility to the system will be high when tactics have long deadlines since large numbers of offers will be exchanged. Consequently, the agents will gain relatively less cost adjusted utility.*

We define an environment with long term deadlines as one in which the values of t^c_{max} and t^s_{max} are sampled within $[30, 60]$ ticks of a discrete clock. Note that we allow $t^c_{max} \geq t^s_{max}$ and $t^c_{max} < t^s_{max}$. Since high values of κ^a overconstrain the true behaviour of tactics, we set $\kappa = 0.1$ for both agents. We then run the experiments in two hundred environments. In each environment the client begins the bidding process.

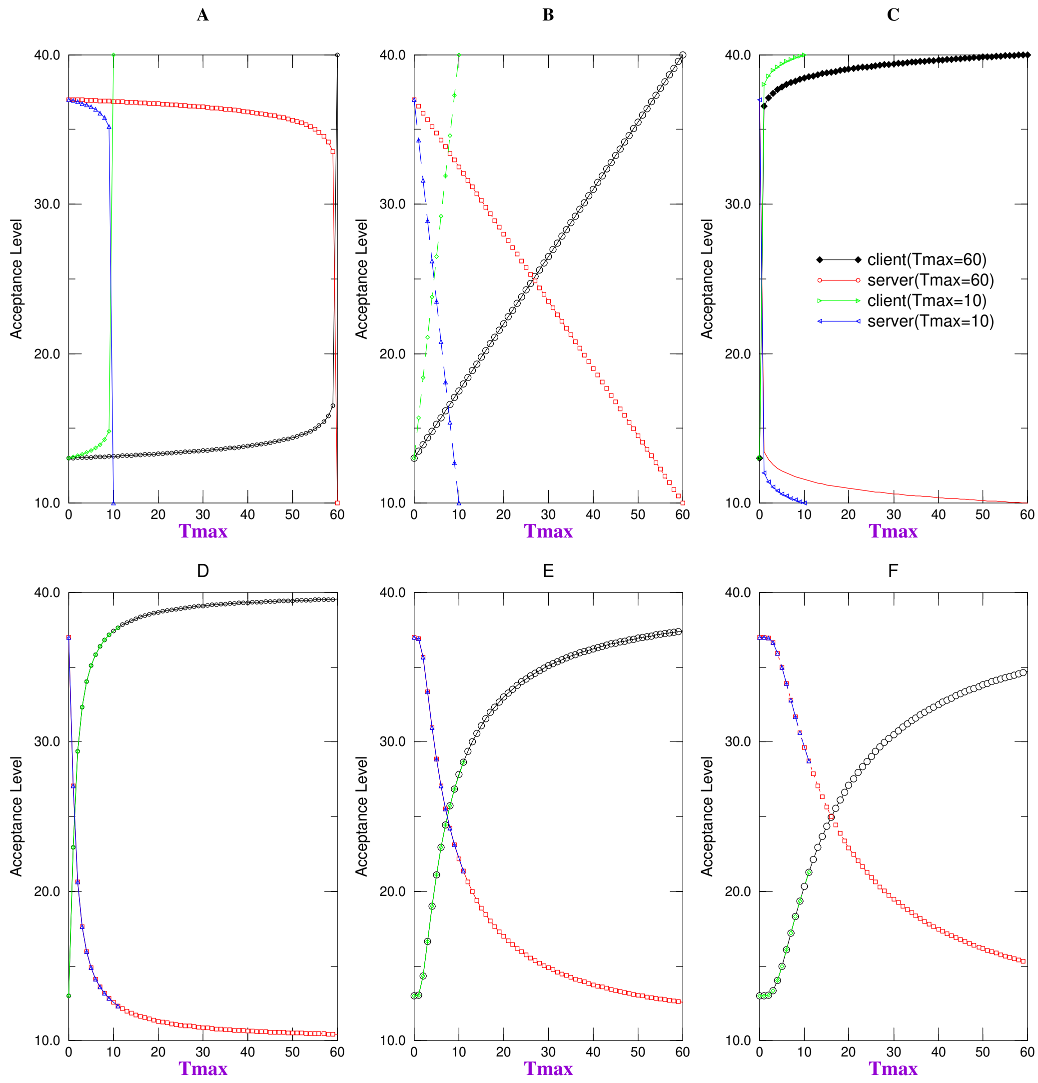

Fig. 5. Behaviour of pure tactics when $\kappa = 0.1$, $min^c = 10, max^c = 40$ and $min^s = 10, max^s = 40$ for $t^s_{max} = 10$ $t^c_{max} = 60$. A) Boulware, $\beta = 0.1$, B) Linear $\beta = 1$, C) Conceder $\beta = 20$, D) Thread1 $\mu = 1$, E) Thread2 $\mu = 6$, F) Thread3 $\mu = 13$

Our first observation is that there is a difference in intrinsic utility between the client and the server, with the client gaining more utility than the server for *all* tactics (Figure 6.A). This is an unexpected result because we believed that given sufficient numbers of sampled environments a client agent should, on average, gain the same utility as a server agent for a given tactic. To explain this difference we hypothesised that the observed difference in the utilities is caused by the way relative-titfortat and average-titfortats compute their offers. These latter tactics compute their next offer based on the product of the quotient between counter offers of the opponent and the previous offer the agent (section 4.3). Consider, as an example, an offer of *£100* and *£1000* as the previous offer of a client and a server respectively. A concession of 20% has

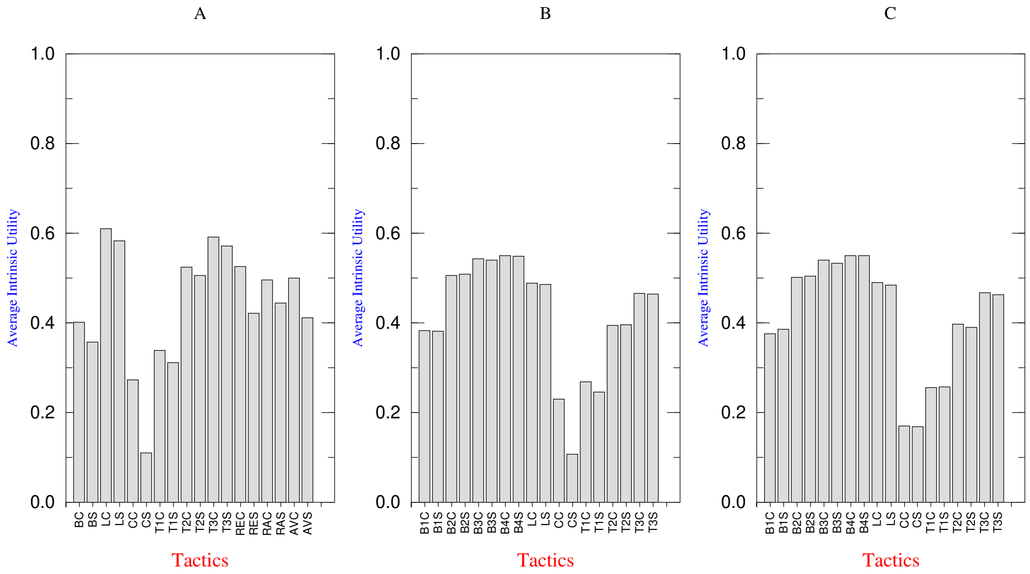

Fig. 6. Average intrinsic utility, A) client first, with behaviour-dependents B) client first, without behaviour-dependents, C) random choice of first bidder, without behaviour-dependents.

the value of *£20* for the client and *£200* for the server with the result that this concession "pulls" down the server's offer faster than "pushing" the client's offer up. We tested this hypothesis by removing the behaviour-dependent tactics from the experimental groups. We replaced them with three additional time-dependent tactics with $\beta \in \{0.25, 0.4\}$, $\beta \in \{0.4, 0.7\}$ and $\beta \in \{0.7, 0.9\}$ (Boulware2, Boulware3 and Boulware4 respectively) whose behaviour becomes progressively linear with increasing values for β. As figure 6.B shows, this did indeed reduce the differences between the client and the server.

However, significant differences remain between the client and the server for tactics which are conciliatory (i.e Conceder and Thread1). We hypothesised that the reason for this residual difference is due to the order of who begins the negotiation process. The execution control in our experimental procedure means that the client always starts the negotiation at $t = 0$, followed by the server at $t = 1$, and so on. For the client the value of $\alpha^c_{price}(0) = \kappa^c_{price}$ (figure 5.C and section 4.1). However, the server's first offer is based on $\alpha^s_{price}(1)$ and when $\beta > 1$ the polynomial almost concede to the reservation at the $t = 1$. Therefore a client who starts the negotiation process always reaches $\alpha^c_{price}(t^c_{max}) = 1$ before the server. The same is true for the behaviour of Thread1. To test this hypothesis we changed the experimental procedure to randomly select who goes first in each environment. As Figure 6.C shows, the utility between a client and a server is equal if we randomly select who goes first. For the remainder of the experiments, we re-introduce the behaviour-dependent tactics into the population and randomly select who goes first.

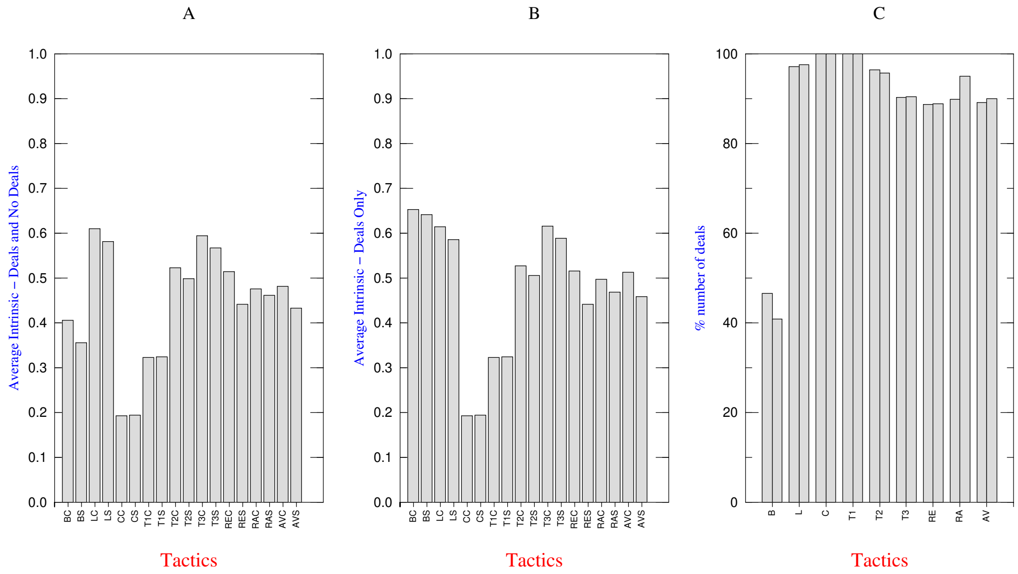

Fig. 7. A) Average intrinsic utility for both deals and no deals, B)Average intrinsic utility for deals only, C) Percentage of deals made.

Having refined our experimental procedure, we return to the task of tactic evaluation. Figure 7.A shows the intrinsic agent utility for all tactic pairs (which includes the utility for both deals and no deals), while figure 7.B is the same measure but only for the cases where a deal was made (see section 6.2.1). Finally, figure 7.C shows the percentage of successful deals for tactic pairs. We predicted that a tactic which approaches reservations at the slowest rate (i.e a Boulware) should rank the highest in average utility. However, from figure 7.A we observe that the actual dominant tactic when there is plenty of time for negotiation is a linear type of tactic (Linear, Thread3 and Thread2), followed closely by behaviour-dependent tactics. Note, imitative tactics never do better than other tactics; the best they can do is to gain equal utility as the best tactic [1]. To help explain this deviation from our predictions, we note that the Boulware tactics make significantly fewer deals than all the other tactic families (figure 7.C). Taking this into account, we examined the average intrinsic utility for cases in which deals are made (figure 7.B). This shows that when Boulwares do make deals, they do indeed receive a high individual utility. From these observations and the data from figure 6.C, we conclude our initial hypothesis does not hold because of the composition of the tactic population. We predicted that in an environment which there is plenty of time to reach a deal, Boulware should rank higher than tactics that approached reservation values quickly. However, the combined observations from figure 6.C and figure 7.A suggest that there exists a limited range of values for the rate of change of acceptance levels that performs best. The Boulware tactics in 7.A are represented by $\beta \in \{0.01, 0.2\}$ which gives rise to very rigid tactics which rarely change their offers. However, as Boulware becomes less fixed and more Linear, as in figure 6.C, they are more successful.

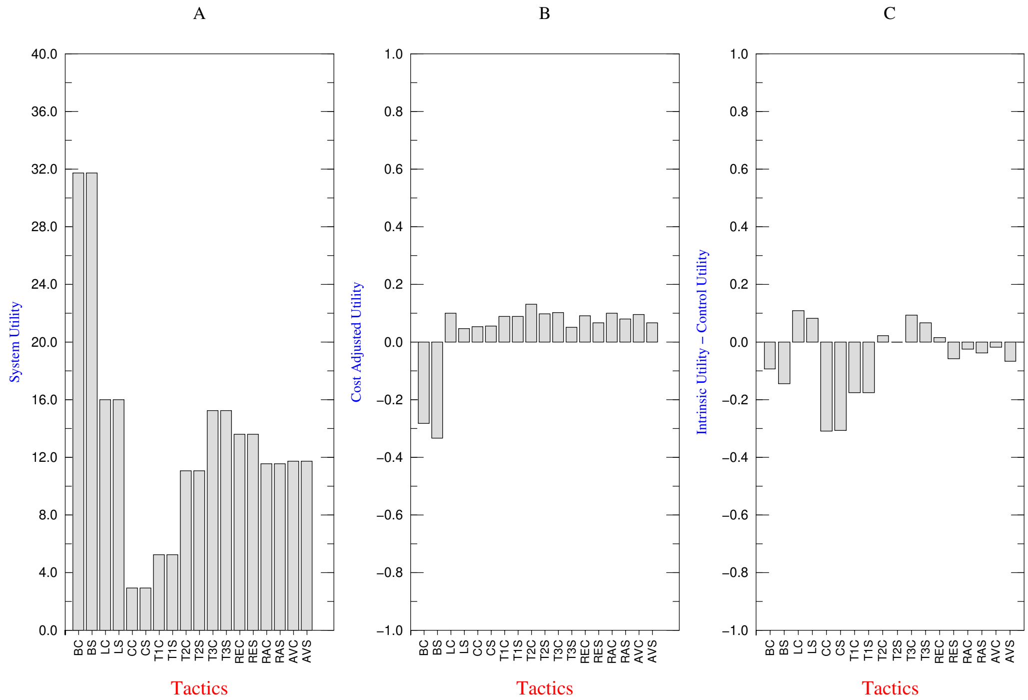

Fig. 8. A) Average System Utility, B)Average Cost Adjusted Utility, C) Comparisons to Control.

Figure 8 shows the outcomes of the experiments with respect to system utility (8.A), cost adjusted benefits (8.B) and control measures (8.C). Our hypothesis with respect to the system utility is confirmed; the tactic that uses the most system resource is Boulware, and the least is the conceder family. In addition, although a Boulware tactic has a higher intrinsic agent utility than conceder type tactics (Conceder and Thread1 figure 7.A), when the the cost of communication is taken into consideration the converse is true. The cost adjusted utilities of other tactics have similar magnitude. The reason for this is that the cost adjusted benefit, which is the product of the intrinsic utility and a function of the number of exchanged messages (see 6.2.2), is sensitive to large fluctuations in the product and assigns similar utilities to non-extreme values.

Finally, we observe that the comparison of our tactics with respect to the controls follows the same broad pattern as the intrinsic agent utility. Linear type tactics (Linear, Thread1 and Thread3) perform better than the controls and the Conceder types (Conceder and Thread1) perform worst. This is to be expected since the closer the tactic's selected deal to the Pareto optimal deal (intrinsic utility of 0.5), the closer to zero the differential between the intrinsic utility and the control utility becomes. As we can see from figure 7.A, the only tactics which approach or exceed an average intrinsic utility of 0.5 are linear type tactics, (namely Linear, Thread2 and Thread3), followed by the behaviour-dependents, then Boulware and, finally, conceder type tactics.

6.3.2 Short Term Deadlines

A change in the environmental setting can mean a radical change in the successfulness of a particular family of tactics. Therefore we carried out an experiment to investigate the behaviour of tactics in environments where the deadlines are short (cf. 6.3.1). For this case, our hypotheses are:

Hypothesis3: *When there is a short time frame to negotiate, tactics which quickly approach their reservation values will make more deals and gain relatively more intrinsic utility than tactics whose rate of approach to the reservation is slow.*

Hypothesis4: *Since deadlines are short, the number of messages exchanged to reach a deal will be small. Consequently the system utility will be low and the cost adjusted utility will be high.*

Short term deadlines are obtained by sampling a pair of values for $[t^c_{max}, t^s_{max}]$ between a low range of values (in this case between $[2, 10]$ ticks of a discrete clock). For the reasons outlined above, we set $\kappa = 0.1$ for both agents and randomly select who starts the bidding process. Again, averages are computed over two hundred environments.

Figure 9 shows the results obtained for these experiments. The first observation is that for most tactics the overall intrinsic utility, the system utility and the number of deals made (figures 9 A, C and B respectively) are significantly lower than the respective measures for the long deadline experiments. A lower system utility is expected since far fewer messages can be exchanged in the allocated time. Also, because fewer messages are exchanged the agents pay less tax and, consequently, keep more of the derived utility. The other measures require further analysis.

With long term deadlines, most tactics, apart from Boulware, were making deals approximately, 90% to 95% of the time, whereas with short term deadlines only the Conceder is making similar, although reduced, percentage of deals. This reduction is due to tactics that are either insensitive to changes in their environment or have a slow rate of approach to reservation values (i.e Thread3, Thread2 and Boulware respectively - see figure 5). All other tactics consequently fail to make many deals when interacting with these tactics. Because the length of the thread is independent of the time deadlines the resource-dependent tactics cannot distinguish between short term and long term deadline. This claim is supported by the observation that Thread1 gains equivalent intrinsic utility independently of deadlines (figure 9A). Furthermore, resource-dependent tactics are differentiated with respect to μ, the amount of time an agent considers reasonable to negotiate with another agent. If an agent does not reason about time deadlines *and* erroneously assumes a value for μ which is close to or above t_{max}, then it will be unsuccessful in envi-

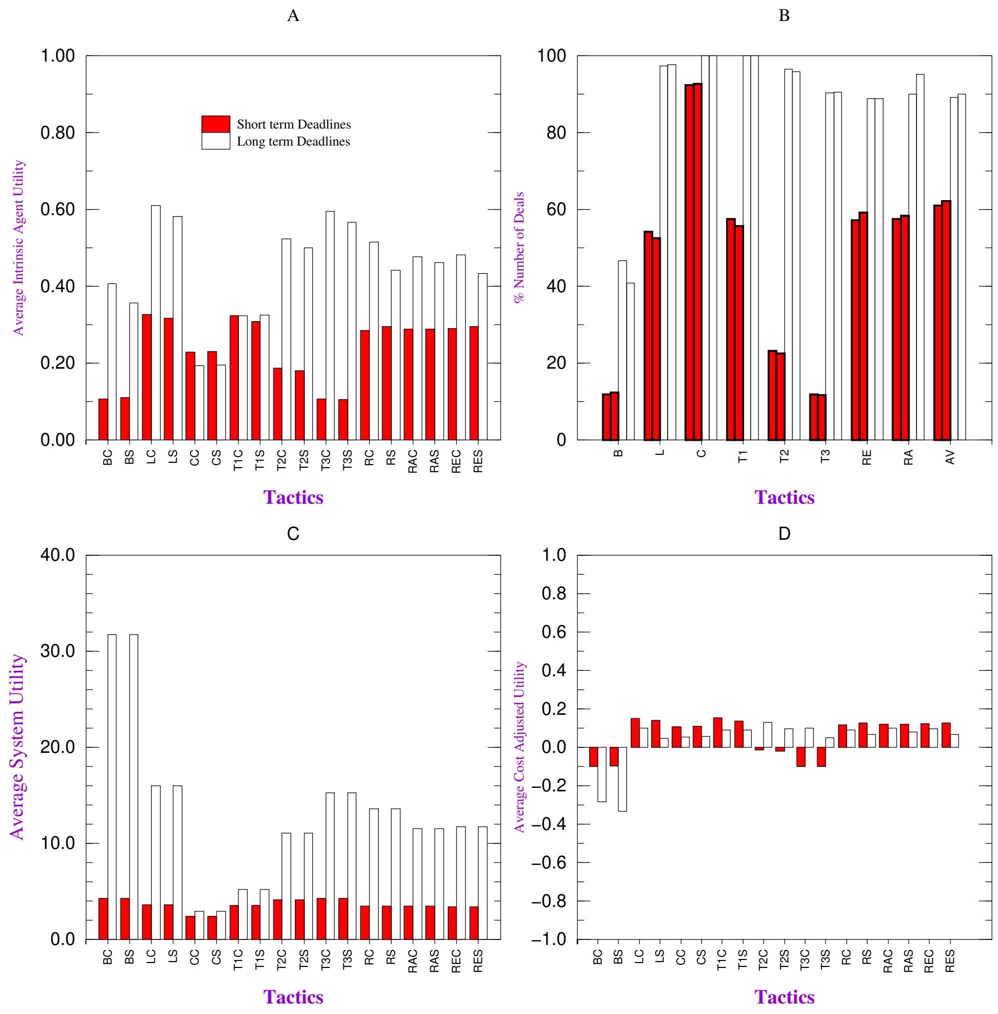

Fig. 9. Comparative data for long and short term deadlines. A) Average Intrinsic Utility, B) Percentage Number of Deals C) Average System Utility, D) Average Cost Adjusted utility.

ronments where deadlines are important. The relatively low intrinsic utility of Thread3 and Thread2 (ranked 9th and 7th respectively - figure 9A) in short term deadline supports this. When deadline is long then resource-dependent tactics with $\mu > 1$ gain large intrinsic utility because they approach reservation values in a linear way. However, the same behaviour in short term deadlines is less beneficial.

Hypothesis three is supported by the relative reductions in intrinsic utility for Boulware, Thread2 and Thread3 and an increase for Conceder. Whereas in long term deadlines Boulware, Thread2 and Thread3 ranked higher than Conceder the reverse is true for short term deadlines. In short term deadlines, tactics that quickly approach reservation values gain higher intrinsic utility than Boulware type tactics.

Again, we discovered that the dominant tactic is a linear type (i.e Linear and to some extend Thread1) suggesting that the best tactic, independently of time deadlines, is a tactic that approaches reservation values in a linear fashion. The behaviour-dependent tactics also gain relatively high utilities in both cases ranking approximately third and fourth for short and long term deadlines respectively. Thus whereas most tactics have large fluctuations in rankings across environments, the behaviour-dependent tactics maintain a stable position, indicating their general robustness and usefulness in a wide range of contexts.

6.3.3 Initial Offers

In our model an agent's reservation values are private. This means that no other agent has any knowledge of where in the range of acceptable values an opponent begins it's bidding process, nor where it is likely to end. Given this constraint, an agent must decide where in it's reservation ranges it must begin *it's* negotiation offers: What should be the value of κ^a in the face of this uncertainty? To help answer this question we formed the following hypothesis [15]:

> **Hypothesis5:** *When the deadline for agreements is not short, making initial offers which have values near the maximum of V^a_{price} leads to deals which have higher intrinsic agent utilities than initial offers near the minimum of V^a_{price}. In other words, a server that starts bidding close to max^s_{price} is more likely to end up with deals that have a higher utility than a server who starts bidding close to min^s_{price}. The converse is true for the client.*

To test this hypothesis, we let both agents have reasonably long deadlines, $t^c_{max} = t^s_{max} = 60$, and made κ^c a constant at 0.1 so that we only had a single independent variable, κ^s, which we sampled between the values $[0.01, 0.2]$ for high initial price offers and $[0.8, 0.99]$ for low initial price offers. The client is being cautious in it's first offer. All other environmental variables were chosen as in previous experiments.

Figure 10 confirms our prediction that a server which begins bidding at values near the maximum of V^s_{price} (figure 10.A) has a higher average intrinsic utility than a server that begins bidding at values near the minimum of V^s_{price} (figure 10.B). If κ^s is close to κ^c (the client starts bidding at low values and the server begins with high offers) then both agents gain equivalent utility and take many rounds of negotiations before a deal is found (10.C) because tactics begin their negotiation at some distance from the point in time where bids have values which are mutually acceptable.

[15] Note: V^s_{price} increases and V^c_{price} decreases with increasing price offers.

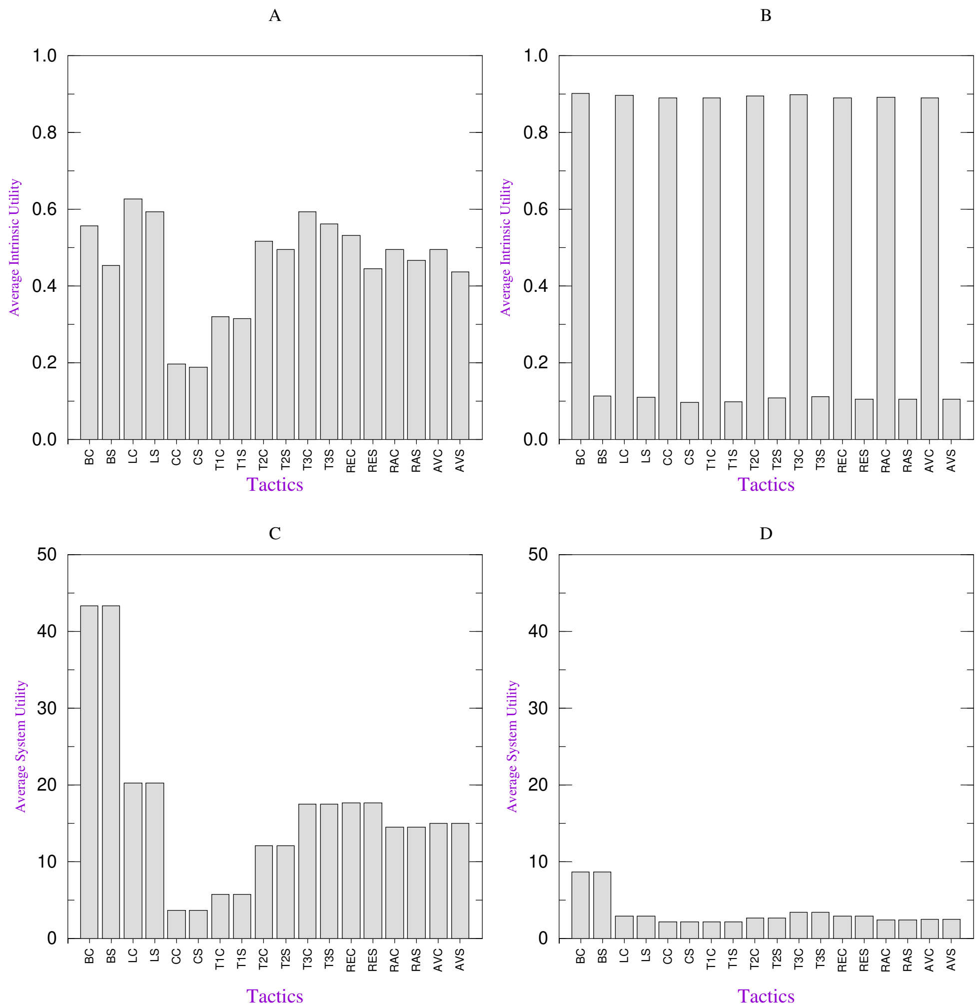

Fig. 10. A) Average Intrinsic Utility for $\kappa^s \in \{0.01, 0.2\}$, B) Average Intrinsic Utility for $\kappa^s \in \{0.8, 0.99\}$, C) Average System Utility for $\kappa^s \in \{0.01, 0.2\}$ and D)Average System Utility for $\kappa^s \in \{0.8, 0.99\}$. $\kappa^c = 0.1$.

Conversely, if κ^s is not close to κ^c (both the client and server start bidding at low values), then the client benefits more and the server benefits less. This is because the initial offers of the server are now immediately within the acceptance ranges of the client (confirmed by the number of messages exchanged before a deal is reached (figure 10.D). Thus the client gains relatively more utility than a server since the initial offers of both agents are low and deals are made at low values [16].

We can further explain the influence of κ on the behaviour of tactics from the observations shown in figure 11. κ^a is used by all tactics for generating the initial offer but, for exposition purposes, we will only discuss the results with

[16] When κ^s is distinctly different from κ^c there is little differention among intrinsic utilities. This is why we set $\kappa^a = 0.1$ for both agents in section 6.3.1 and 6.3.2.

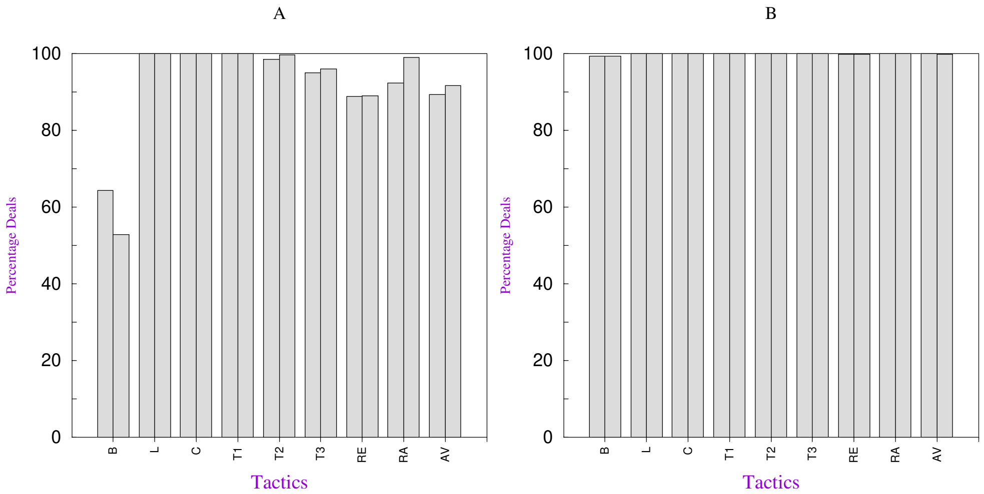

Fig. 11. Percentage of Successful Deals when $\kappa^c = 0.1$ and A) $\kappa^s \in \{0.01, 0.2\}$, B) $\kappa^s \in \{0.8, 0.99\}$.

respect to the Boulware tactic family (since this offers the greatest difference in behaviour). When κ^s is low, Boulwares have a lower percentage of deals relative to other tactics (figure 11.A). Conversely, when κ^s is high then Boulware almost equal all other tactics in the percentage of deals they make (figure 11.B). This is because at low values of κ^s the shape of the acceptance levels for a Boulware is almost a step function and a straight line near to or at min^s when κ^s is high. A server playing a Boulware tactic makes a small number of high utility deals when the acceptance levels tend towards being a step function (compare figures 11.A and 10.A), but makes larger number of lower utility deals when the acceptance level is almost a straight line (figures 11.B and 10.B). Therefore, as the value of κ increases, the likelihood of a deal increases, but the utility of the deal decreases.

7 Related work

Research in negotiation models has been pursued in different fields of knowledge: game theory, social sciences and artificial intelligence. Each field has concentrated on different aspects of negotiation, making the assumptions that were pertinent for the goal of their study. In game theory, researchers have been interested in mechanism design: the definition of protocols that limit the possible tactics (or strategies) that can be used by players. For instance they are interested in defining protocols that give no benefit to agents that misrepresent or hide information [11]. In this work disclosure of information is acceptable, because by doing so it will benefit the agent in finding an optimal solution for itself. Contrary to our model, and as we discussed in Section 2, this is an inappropriate assumption from the point of view of real applications. As has been argued elsewhere [16], these and other assumptions limit the applicability of game theory models to solve real problems. In a paper in this issue Wellman and Wurman present a justification of the adaptation of mechanism design to situations in which disclosure of information is not possible or acceptable [14]. They present market price systems as institutions that can be used to model resource allocation in general. Our approach agrees with this point of view concerning disclosure of information and complements it in that we concentrate more on the internal decisions of negotiating agents given a particular protocol, and not on the process of mechanism design.

Our interests lie in investigating the process of negotiation among agents and not only on the outcome. Hence, our study, and those forthcoming, are much more in the experimental line of [5]. Although we do not concentrate on learning, some similarities can be found with the formalism by Zeng and Sycara [16]. We have not concentrated however on the description of negotiation protocols that has been an important focus of attention for the community of distributed artificial intelligence (see [8] for extensive references).

8 Discussion and future work

This paper presented a formal model of an autonomous agent's decision function as it relates to the process of service-oriented negotiation. The model defines a number of tactics which agents can employ during negotiations and it indicates how an agent can change these tactics over time to give various forms of strategic behaviour. The form of the model, and the assumptions it makes, has been guided by our experiences in developing real-world agent applications for the domain of business process management. For this reason, the model is well suited for practical agent applications.

In earlier work [13], we proved that agents negotiating using our model were guaranteed to converge on a solution in a number of well defined situations. In this paper, we sought to extend these results and evaluate the model in a wider range of circumstances. To this end, we defined a number of basic hypotheses about negotiation using our model and sought to validate them empirically. In particular, with respect to tactics we discovered that: (i) irrespective of short or long term deadlines it is best to be a linear type tactic, otherwise an imitative tactic; (ii) tactics must be responsive to changes in their environment; and (iii) there is a tradeoff between the number of deals made and the utility gained which is regulated by the initial offers.

The aforementioned results confirmed (and rebuted!) a number of basic predictions about negotiation using our model. Our aim for the future is to extend this evaluation to cover a wider range of phenomena. In particular, we intend to: (i) extend the analysis to other types of environment classes (for example, we predict that an increase in the number of agents will affect resource-dependent tactics and dramatically influence the dynamics of all tactic interactions); (ii) investigate the effects of strategies, weighted combination of tactics may outperform pure tactics in certain environments; (iii) investigate the tactic "pool" which makes up the population, we predict that the number and value of deals made between members of a society that is made up solely of Boulwares will be significantly different to societies where the population has a mixture of Boulwares and imitators. Finally, to gain further explanatory power, we intend to analyse the behaviour of tactics, in these and future environments, at the level of pairwise interactions.

9 Acknowledgements

This project has received the support of the DTI/EPSRC Intelligent Systems Integration Programme (ISIP) project ADEPT, Nortel Technology and the Spanish Research project SMASH (CICYT number, TIC96-1038-C04001).

References

[1] R. Axelrod. *The Evolution of Cooperation.* Basic Books, Inc., Publishers, New York, USA., 1984.

[2] B. Crabtree and N. R. Jennings, editors. *The Practical Application of Intelligent Agents and Multi-Agent Technology*, London, UK., 1996.

[3] G. Debreu. *Theory of Value: An Axiomatic Analysis of Economic Equilibrium.* Wiley, New York, 1959.

[4] E. Ephrati and J.S. Rosenschein. *Multi-Agent Planning as Search for a Consensus that Maximizes Social Welfare*, pages 207–226. Lecture Notes in Artificial Intelligence 830. Springer-Verlag, 1994.

[5] A. Ito and H. Yano. The emergence of cooperation in a society of autonomous agents - the prisoner's dilemma game under the disclosure of contract histories. In Victor Lesser, editor, *Proceedings of the First International Conference on Multi-Agent Systems*, pages 201–208, San Francisco, USA, 1995. AAAI Press/The MIT Press.

[6] N. R. Jennings, P. Faratin, M. J. Johnson, T. J. Norman, P. O'Brien, and M. E. Wiegand. Agent-based business process management. *Int Journal of Cooperative Information Systems*, 5(2–3):105–130, 1996.

[7] S. Kraus, J. Wilkenfeld, and G. Zlotkin. Multiagent negotiation under time constraints. *Artificial Intelligence Journal*, 75(2):297–345, 1995.

[8] H.J. Mueller. Negotiation principles. In G. M. P. O'Hare and N. R. Jennings, editors, *Foundations of Distributed Artificial Intelligence*, Sixth-Generation Computer Technology Series, pages 211–229, New York, 1996. John Wiley.

[9] D. G. Pruitt. *Negotiation Behavior*. Academic Press, 1981.

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

[11] J. S. Rosenschein and G. Zlotkin. *Rules of Encounter*. The MIT Press, Cambridge, USA, 1994.

[12] S. Russle and E. Wefald. *Do the Right Thing*. The MIT Press, 1991.

[13] C. Sierra, P. Faratin, and N. R. Jennings. A service-oriented negotiation model between autonomous agents. In M. Boman and W. Van de Velde, editors, *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, pages 17–35. Springer-Verlag, 1997.

[14] M. P. Wellman and P. R. Wurman. Market-aware agents for a multiagent world. *Robotics and Autonomous Systems*, This issue, 1998.

[15] L. A. Zadeh. Fuzzy sets. *Information Control*, 8:338–353, 1965.

[16] D. Zeng and K. Sycara. How can an agent learn to negotiate. In J. Mueller, M. Wooldridge, and N. Jennings, editors, *Intelligent Agents III. Agent Theories, Architectures, and Languages*, number 1193 in LNAI, pages 233–244. Springer Verlag, 1997.