Using Similarity Criteria to Make Issue Trade-Offs in Automated Negotiations

P. Faratin
Laboratory for Computer Science
Massachusetts Institute of Technology Cambridge, MA 02139, USA.
peyman@mit.edu

C. Sierra
IIIA-CSIC,
Campus UAB, Bellaterra,
08193 Barcelona, Spain.
sierra@iiia.csic.es

N. R. Jennings
Dept. of Electronics and Computer Science,
University of Southampton,
Southampton SO17 1BJ, UK.
nrj@ecs.soton.ac.uk

Abstract

Automated negotiation is a key form of interaction in systems that are composed of multiple autonomous agents. The aim of such interactions is to reach agreements through an iterative process of making offers. The content of such proposals are, however, a function of the strategy of the agents. Here we present a strategy called the *trade-off* strategy where multiple negotiation decision variables are traded-off against one another (e.g., paying a higher price in order to obtain an earlier delivery date or waiting longer in order to obtain a higher quality service). Such a strategy is commonly known to increase the social welfare of agents. Yet, to date, most computational work in this area has ignored the issue of trade-offs, instead aiming to increase social welfare through mechanism design. The aim of this paper is to develop a heuristic computational model of the trade-off strategy and show that it can lead to an increased social welfare of the system. A novel linear algorithm is presented that enables software agents to make trade-offs for multi-dimensional goods for the problem of distributed resource allocation. Our algorithm is motivated by a number of real-world negotiation applications that we have developed and can operate in the presence of varying degrees of uncertainty. Moreover, we show that on average the total time used by the algorithm is linearly proportional to the number of negotiation issues under consideration. This formal analysis is complemented by an empirical evaluation that highlights the operational effectiveness of the algorithm in a range of negotiation scenarios. The algorithm itself operates by using the notion of fuzzy similarity to approximate the preference structure of the other negotiator and then uses a hill-climbing technique to explore the space of possible trade-offs for the one that is most likely to be acceptable.

keywords: Multi Agent Systems, Automated Negotiation, Fuzzy Similarity, Trade-off Algorithm

1 Introduction

Automated negotiation is a key form of interaction in systems composed of multiple autonomous agents. It is so important because the agents are autonomous (that is, they decide for themselves what actions they should perform, at what time, and under what terms and conditions [20]) and can have conflicting preferences over state of the world. Given the facts that such agents have no direct control over one another and there are often interdependencies between their actions, conflicts need to be resolved by the process of making proposals and/or trading offers, with the aim of finding a mutually acceptable agreement. In short, by negotiating. More specifically, we view negotiation as *a bargaining process by which a joint decision is made by two parties. The parties first verbalise contradictory demands and then move towards agreements.*

The prevalence and importance of automated negotiation can be seen in the large number of proposed models [8, 18]: ranging from auctions in which the agents' pricing decision problem is solved through showing the dominance of a truthful bidding strategy [57], to models in which the agents' argue for positions and aim to persuade their opponents of the value of particular actions [37]. In this work we are interested in conflicting preferences over complex multi-dimensional decision problems involved in the bi-lateral resource allocation negotiation of *services* [50]. In such duopolistic negotiations, one producer and one consumer have to bargain and come to a mutually acceptable agreement over the terms and conditions under which the producer will execute some activity (service) for the consumer.[1] Specific decision variables that typically need to be *mutually* agreed include the price of the service, the time at which it is required, the quality of the delivered service, and the penalty to be paid for reneging upon the agreement.

The generative model of bargaining presented here shares with other mechanism design models the explicit design of protocols whose execution is a function of the agent's strategy [3]. A protocol is a set of "rules of encounter" [43] between the negotiation participants; that is, who can say what, to whom, at what time. Given a protocol, an agent strategy then defines the model that the individual participants apply to act in line with the protocol in order to achieve their negotiation objectives. However, the goals motivating the design of the protocol and strategy in this work are different from those of classic mechanism design. The latter are more interested in solving the strategic mis-representation problem that occurs whenever agents have an incentive to mis-represent their true preferences in order to maximise their own utility. A mechanism design solution to this problem consists of centrally designing direct incentive compatible or strategy proof decision rules that have certain properties [29, 43]. Although we acknowledge strategic misrepresentation is a concern in multi-agent systems, we are also interested in the types of decision problems that are not only highly complex in dimensionality (rather than simply dividing the cake) but that also place bounding limits on the performance of the agent by the virtue of their complexity. Indeed, the combination of these two factors can lead to sub-optimal decisions, thereby threatening classic solution concepts from mechanism design [8]. Therefore, we make the implicit assumption that social agreements to complex problems are achieved through an iterative and indirect fashion similar to real world bargaining where ill informed players interact and communicate to reach a social choice. These protocol and agent assumptions were necessary in order to design a negotiation system for the types of real world problems we have been involved in: business process management [19], telecommunications network management [9], and e-commerce [35, 42]. These assumptions are as follows. Firstly, agents have only limited information about their negotiation opponent. Although a mechanism can theoretically be designed to incentivise

[1] It is now common practice for organisations to view their function in terms of the services that they provide to their various stakeholders. Thus, a service-oriented view, and by extension service-oriented negotiation, should be seen as covering a wide spectrum of possibilities.

agents to truthfully reveal their preferences to a central planner, it is assumed that such a task is highly costly for high dimensionality problem tasks. Instead, we are interested in a distributed approach where solutions are sought when agents do not know the other player's preferences for negotiation outcomes, their reservation values, or their resource constraints. Secondly, agents are not computationally unbounded. Computation, informally defined as search, is costly in both time and resources. Thirdly, agents are engaged in a multi-criteria decision problem modeled as multi-dimensional contracts that include both continuous and discrete decision variables. Finally, due to the uncertainties in interaction, the complexity of the computation involved in dealing with multi-dimensional goods, and the presence of boundedness, the depth of the game tree is implicitly managed by assuming a finite horizon of interactions. These interactions also follow the rules of an alternating sequential protocol in which the agents take turns to make offers and counter offers [45]. The protocol terminates when the agents come to an agreement or when one of them withdraws from the negotiation.

One implication of the above assumptions is that it is not possible to pre-compute an optimal negotiation strategy at design time. Rather the agents need to adopt a heuristic and satisficing approach for their strategy [8, 23]. This is in contrast with deductive models of negotiation where each agent explicitly represents and reasons with the decision tree of the entire game [30, 15]. In this case, a negotiation strategy is then the specification (using for example backward induction [2]) of a sequence of choices for every decision node in the game tree, with the property that *both* the final choices and the complete sequence (sub-game) of choices are often in equilibrium [45]. However, because representation and reasoning under such a system can be computationally intractable [23, 25, 36] we have been involved in developing *approximating* decision models for a more limited type of agent that has no explicit representation of the entire game tree. Then, rather than computing the best response given knowledge of the end tree, an agent uses the information gained sequentially in interactions to heuristically form a prediction of the future based on the history of the interaction so far. Decision making in an intelligent negotiating agent can be supported by any number of heuristics that assist it in searching for potential deals. In the decision model presented in this paper the reasoning process of an agent at each sequence of the negotiation is characterised as meta deliberation over the execution of either a concessionary or a trade-off mechanism or both. The former mechanism models iterative concession over the score of a contract based on environmental factors such as the time remaining until the deadline, the amount of resources consumed in the negotiation, and the behaviour of the negotiation opponent (for this reason this is called a *responsive mechanism* since agents react to their prevailing environmental context [12]). This exchange of offers and counter-offers continues until a crossover occurs between the demands of the two agents or one of the agents withdraws. Conversely, reasoning in the trade-off mechanism (described fully below) is characterised by a heuristic function that maps the current demand and the previous offer to a new offer.

In this case, however, such meta decisions are taken not over the whole game tree structure, but rather at *each* decision node of the decision tree that represents only the agent's local optimization problem and not the joint optimisation problem of the dyad. Given this, the goal of this paper is to demonstrate the value of incorporating one heuristic, the similarity heuristic, in the trade-off decision mechanism for a given set of conditions. Additionally, since the strategy of the agent is not under the control of the system designer, we would like to show that rational agents are motivated to implement such a heuristic when faced with uncertainty about their opponents' utility function. However, at the same time we note that the computational and representational simplicity of a heuristic approach is traded-off against our inability to predict or specify equilibrium strategies, since agents do not explicitly represent and reason about the choices of the other agent. Furthermore, since heuristics can fail we are forced to accept the possibility of failing to find better decision nodes with higher objective values.

However, although multi-dimensional decision problems introduce additional computational complexities, they nonetheless present inherent opportunities for increasing the social welfare of the deal through trading off between decision variables. This opportunity has been the motivating factor for developing the heuristic model presented in this paper. In our previous work we reported on a concessionary strategic mechanism for assigning values to decision nodes [12]. However, this responsive mechanism fails to explore the space of potentially jointly better solution nodes because it cannot explore different possible value combinations for the local negotiation decision variable. Thus, for example, a contract in which the service consumer offers to pay a higher price for a service if it is delivered sooner, may be of equal value to the consumer as one that has a lower price and is delivered later. However from the service provider's point of view, the former may be acceptable and the latter may not. The original model does not allow the agents to explore for such possibilities because it treats each decision variable independently and only allows agents to concede on decision variables (thus producing a contract of lower value to themselves).

To overcome this limitation and to increase the efficiency of deals, agents need the ability to make *trade-offs between negotiation decision variables.* Intuitively, a trade-off is where one party lowers its value on some negotiation decision variables and simultaneously demands more on others. Thus, an agent may accept a service of lower quality if it is cheaper or a longer deadline if it receives a higher quality. Such movements are intended to generate an offer that, although of the same value to the proposer, may be of greater benefit to the negotiation opponent. This, in turn, should make agreement more likely and increase the overall *joint gains* [41] between the two agents. The particular heuristic we consider in this paper is based on the degree of *similarity* between two consecutive choices.

The contribution of this work is twofold. Firstly, current models of automated negotiation have largely ignored the problem of multi-issue negotiation and the additional possibility and challenging problems of making trade-offs between decision variables We aim to rectify this ommission in section 3. Secondly, we present a novel linear algorithm that enables agents to make trade-offs between both discrete and continuous negotiation decision variables, in the presence of information uncertainty and resource boundedness for multi-dimensional goods. The algorithm itself operates by using the notion of fuzzy similarity [64] to approximate the preference structure of the negotiation opponent and then uses a hill-climbing technique to explore the space of possible trade-offs for the one that is most likely to be acceptable. Although the domain of applicability of the algorithm is currently restricted to linear problems, the abstract underlying similarity model itself supports a *component* of the overall negotiation algorithm and can be used by any negotiating agent. Moreover, this algorithm has been analysed theoretically (to determine its complexity) and evaluated empirically (to ascertain its operational performance).

The remainder of the paper is structured as follows. Section 2 investigates the space of negotiation outcomes and outlines where and why trade-offs are possible. Section 3 presents our algorithm for making trade-offs in service-oriented negotiations. Section 4 provides an empirical evaluation of our trade-off mechanism in a range of negotiation scenarios. Section 5 compares our approach to previous work in this area. Finally, section 6 outlines our conclusions and our plans for future work.

2 The Rationale for Making Trade-Offs

This section analyses the range of outcomes that can occur when two agents negotiate with one another. It does so in order to identify why and where trade-offs are possible. In this work, it is assumed that the agents (a and b) have to negotiate a multi-dimensional contract $o \in O$, where O is the set of possible contracts. Figure 1 A shows a simplified two decision variable version of the problem as an

Edgeworth box. For the purpose of exposition, a single contract clause (decision variable) is taken to represent a commodity and (re)assignment of a value to a decision variable as its (re)allocation. Thus, in this figure two agents a and b have to reach a contract over the allocation of two commodities (1 and 2). Furthermore, each agent is assumed to have an initial endowment of both commodities (has made an initial choice over the pair of contract decision variables before negotiation). The initial endowment of a and b is given by $w^a = (w_1^a, w_2^a)$ and $w^b = (w_1^b, w_2^b)$ respectively and is shown in figure 1A as the point $(w_1^a, w_2^a), (w_1^b, w_2^b)$. The dimensions of an Edgeworth box represent the quantities available of the good. No allocation of either good to a or b is represented by O^a and O^b respectively. The general question is then, what allocation of total units of good 1 and good 2 are feasible? In other words, what decisions over the total ranges of all decision variables are feasible? If an allocation to agent a and b over commodities 1 and 2 is given by (x_1^a, x_2^a) and (x_1^b, x_2^b) respectively, then an allocation is feasible *iff* $x_1^a + x_1^b \leq w_1^a + w_1^b$ and $x_2^a + x_2^b \leq w_2^a + w_2^b$. That is, all points in the box, including the boundary, represent a feasible allocation of the combined endowments. However, some allocations will be blocked by one/both agents while others make both agents better. This is because of the agents' preferences. These are shown by the convex indifference preference curves (or iso-curves) of the two agents in figure 1A, where each curve represents the indifference an agent has over the increasing/decreasing utility of one commodity versus the simultaneous decrease/increase in utility of the other commodity. Allocations along the O^a-O^b and conversely O^b-O^a axis are associated with an increasing value for agent a and b respectively.

Given the above, a feasible allocation can be blocked by an agent when an allocation that increases the utility of one decreases the utility of the other. However, the welfare of *both* agents is increased at the point where the convex indifference curves of each agent intersect. A hypothetical set of such points is shown in figure 1A as solid black ovals. These allocations are said to be pareto-optimal over the endowment allocation (a formal definition is given below). Pareto-optimality implies that if agents have an option to opt out of negotiation then the only possible allocations need to be pareto improving allocations. However, since there are a number of pareto-optimal contracts given the endowment, the question remains which will be the one selected. One solution to this indeterminacy problem is to treat the problem as a bargaining problem in a perfectly competitive market where utility maximising agents trade commodities for given announced prices. Prices are then iteratively lowered or increased with excess supply or excess demand respectively, until the market clears at a general equilibrium [5, 39]. The First Fundamental Theorem of Welfare Economics then states that given consumers' preferences are well behaved, trading in perfectly competitive markets implements a Pareto-Optimal allocation of the economy's endowment.

Solutions to this indeterminacy problem have also been attempted in a more axiomatic fashion from game theory where a single solution is selected that satisfies a set of axioms. To show this, the bargaining problem of figure 1A is mapped from the decision variable space to the utility space representation of figure 1B using a utility function $U_i : O \rightarrow [0, 1], i \in \{a, b\}$ [2]. In this figure, utopia corresponds to the situation where both agents obtain their highest aspirational level. If the agents fail to reach any deal, they each receive a conflict payoff. The set of possible outcomes, including utopia (payoffs $(1, 1)$) and the conflict point c (payoffs $(0, 0)$), are shown in figure 1B. The feasible set of outcomes is denoted by B in figure 1B which contains those agreements that are *individually rational* and is bounded by the *pareto optimal* line. An agreement is individually rational if it assigns each agent a utility that is at least as large as the agent can guarantee for itself from the conflict outcome $\mathbf{x_c}$. Pareto optimality is defined for a bargaining game $(B, \mathbf{x_c})$ in the following manner [6].

[2] Such a perfect curvilinear shape is only obtained under the assumptions that the utility functions of the agents are perfectly concave and differentiable.

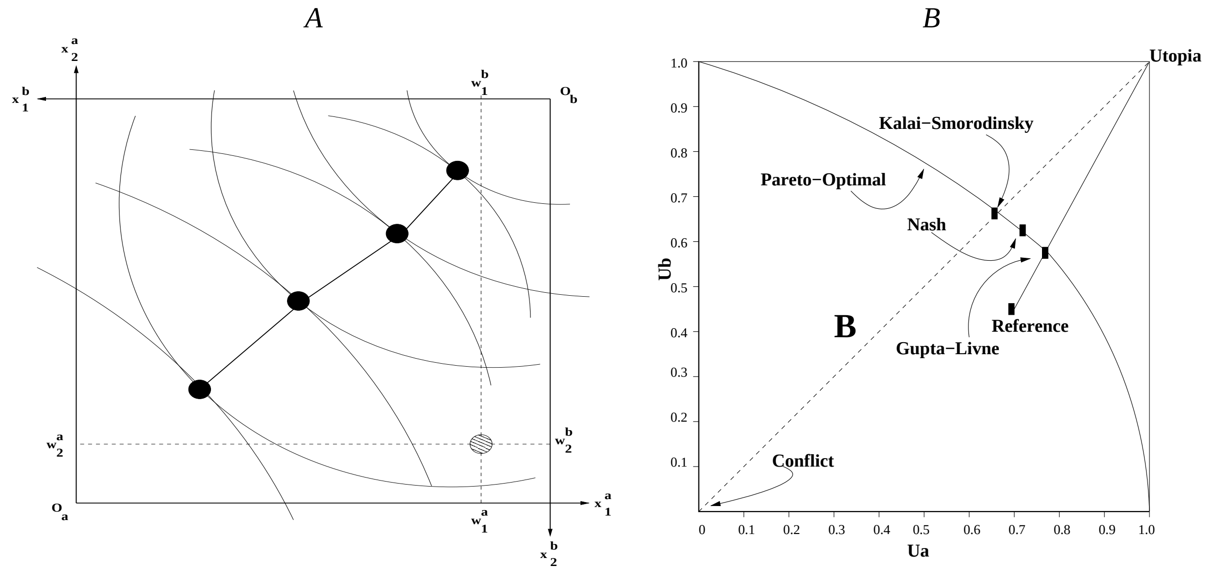

Figure 1: A) Edgeworth Box representing the decision outcome space for a pair (a and b) of negotiating agents. B) Utility outcome space for a pair (a and b) of negotiating agents.

Suppose there are two outcomes $\mathbf{x}$ and $\mathbf{y}$ such that they both belong to the feasible set, $\mathbf{x}, \mathbf{y} \in B$. If $U_i(\mathbf{y}) \geq U_i(\mathbf{x})$, for both a and b, but $\mathbf{y}$ is strictly preferred for at least one agent, $U_i(\mathbf{y}) > U_i(\mathbf{x})$ for $i \in \{a, b\}$, then the outcome $\mathbf{x}$ is not pareto optimal. This is formally represented as a function that given the game defined by the pair B and $\mathbf{x_c}$ does not select $\mathbf{x}$—i.e., $f(B, \mathbf{x_c}) \neq \mathbf{x}$.

Cooperative (or axiomatic) game theory aims to specify axioms that lead to the selection of a single point on the pareto optimal line as the most desirable solution for a given negotiation. The *Nash bargaining solution* is the most popular such solution that selects an individual outcome from the pareto-set (hence it is efficient) that is also the most equitable outcome. The Nash solution is defined as the point that maximises the *product* of the utilities $(U_{aj} - U_{ac})(U_{bj} - U_{bc})$ where U_{ij} is the utility to player i for settlement j and U_{ic} is player i's conflict outcome utility [30]. One interpretation of the Nash bargaining solution is that agents are motivated by equity or proportional cooperation [27]. Another solution concept is the *Kalai-Smorodinsky* [21] which modifies one of Nash's axioms (independence to irrelevant alternatives to individual monotonicty) and is interpreted as endogenously providing more weighting to the "needier" player [27]. However, such axiomatic models are inappropriate for computational purposes because they specify the solution properties and leave the *process* of how to reach these points unspecified. [3] Thus, there are no guidelines for automating the process of how to actually reach these outcomes. Nonetheless, for evaluation purposes we use the focal or reference point [41] (see below for the computational argument why the Nash bargaining solution, and by implication the Kalai-Smorodinsky, is not chosen). This solution point has been extended to an axiomatic *reference outcome* solution proposed by [16]. The focal point is

[3] As will be mentioned later, non-cooperative models, notably the alternating sequential model of Rubinstein [46], do model the selection of the outcome as a process of negotiation, rather than selection of an outcome that satisfies some desirable property. Indeed, under some strict contexts, non-cooperative models implement the Nash bargaining solution when agents' strategies are in equilibrium. However, although we acknowledge the importance of this body of work, we do not claim deductive and rational equilibrium reasoning by our agents for the reasons given above. We note that equilibria can be attained by myopic agents if we adopt a "mass action" [32] or "evolutive" [2] interpretation of equilibria.

often interpreted as a prominent outcome that replaces the conflict outcome and is often expected to have an important bearing on the outcome of negotiation. For example, in multi-issue negotiations the middle point on each issue often becomes the focal point [40] and the negotiators then try together to find other agreements that are better for both. In section 4.3 we use the axiomatic extension of the reference point as the point that is pareto optimal and lies on the line connecting the reference and utopia points (in fact when the reference point is the conflict point then this solution is identical to Kalai-Smorodinsky's solution point). The reference outcome is simply computed as the mid point of each decision variable. This axiomatic solution has been shown to be particularly appropriate for logrolling in integrative multi-issue negotiations [16][4]. Therefore, the property of the solution we seek to optimise is the distance of an outcome to the point lying on the pareto optimal line and connecting the reference point with utopia.

For us, it is this multi-dimensionality of decision problems that permit increasing the social welfare through agents actively *searching* and communicating nodes of the tree of decision trade-offs. This is in contrast to negotiation over a single decision variable (integrative v.s distributed negotiation respectively, [41], figure 2 a and b). In such situations, the opposing nature of service producers and consumers means that the agents' payoffs are perfectly negatively correlated. Thus an outcome that increases the score of one agent decreases the value of the other. Here *all* the possible outcomes lie on the pareto-optimal line. Furthermore, assuming linear conflicting value functions for the negotiation participants, the sum of each outcome is 1 (i.e., it is a *zero-sum game* [13]). In this scenario, the Nash bargaining solution is easily computed as the mid point (and most equitable) of both agents' value function (i.e., at $(0.5, 0.5)$). Given the single decision variable nature of the negotiation, decision variable trade-offs are naturally not possible. More generally, however, the same arguments also hold for multi-dimensional goods in zero-sum games.

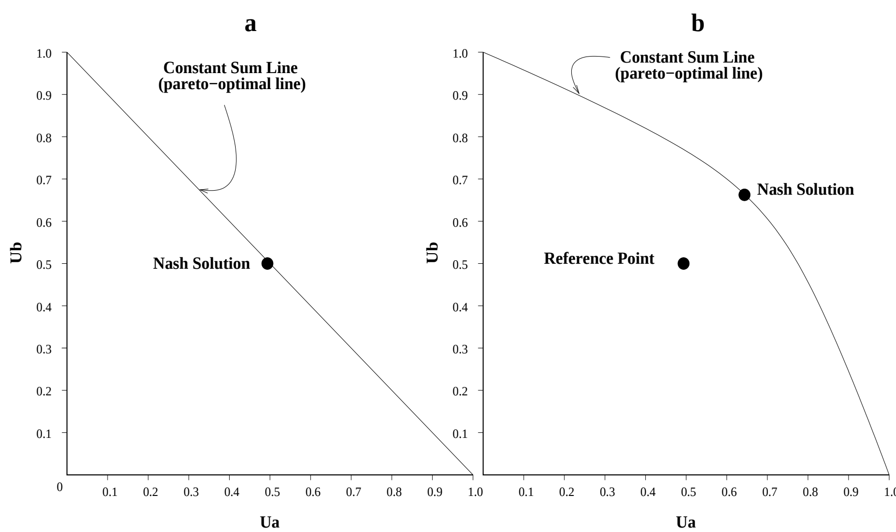

Figure 2: Outcome space for a pair of negotiating agents: a) single decision variable and b) multiple decision variables.

However the games considered in this work are not zero-sum because we can assume that the agents attach different levels of importance to the various negotiation decision variables. Thus, for

[4]The importance of the reference point has also been corroborated empirically [44].

example, one agent may be most concerned with the price of a service, while its opponent may be most concerned with the time by which the service can be delivered. Due to the fact that there are multiple decision variables, each of which has a different importance level, the negotiation outcomes are transformed to a non-constant sum game (where the sum of the utility values along the dimensions of an outcome do not necessarily add up to 1). It is this transformation that opens up the possibility of agents making trade-offs. That is, it is possible to find agreements in which some decision variables are increased in value and others are decreased and overall this will benefit one or both of the negotiation participants simultaneously. In this case, the pareto-optimal line is shown in figure 2b. Now the only points on this line where the sums of the individual values add to 1 is at the point of connection with the x and y axis. Different points along the line then do not necessarily sum to 1 and do not necessarily have the same addition. More importantly, however, it is now possible for negotiation outcomes to lie below the pareto-optimal line (i.e., towards $(0, 0)$) because agents attach different importance weightings to the various decision variables. Consequently, there is scope in the negotiation process to find agreements that are closer to the pareto optimal line, meaning that one or even both of the agents can be better off. This contrasts with the distributive bargaining case where the negotiation outcome *has* to be on the pareto-optimal line (meaning there is no scope for improving one score without decreasing the score of the negotiation opponent). Furthermore, the Nash bargaining solution is no longer at $(0.5, 0.5)$, because the pareto-optimal line has moved from the constant sum line. However $(0.5, 0.5)$ can now be viewed as a reference outcome since it represents the point at which both agents obtain precisely half their aspirational level.

Having defined the outcome landscape and identified the possibility for trade-offs, the next step is to determine how to actually compute such trade-offs. If the agents knew their opponent's preferences and their relative importance weightings, then they could compute solutions that lie on the pareto-optimal line. The regular Nash bargaining solution in fact implements this shared knowledge assumption. However, the Nash bargaining solution is inadequate in cases of multiple decision variables because its computation becomes intractable in the presence of multiple decision variable reservation values and weights. The maximization problem then becomes maximization of a quadratic function with restrictions (the reservation values of a decision variable), where the solution to the quadratic function may violate the restrictions. It is a quadratic problem because the individual utilities of agents are linear:[5]

$$\max \left(\sum_i w_a^i U_a^i(o) \right) \left(\sum_i w_b^i U_b^i(o) \right)$$

However in most realistic situations, this information is simply not available (as discussed in section 1). This means agents need a means of approximating the preferences of their opponent based upon their observable negotiation behaviour. This approximation can then be used to select the outcomes that are closer to, or ideally on, the pareto optimal line. The desired final outcome in the feasible set often depends on the agent's social objectives/goals. These may be to maximise the joint gains of the agents (if they are both from the same organisation) or they may be to increase the value of the agreement to the opponent while keeping their own return constant (if the aim is to find the contract that is most likely to be accepted).

Attempting to approximate the preference structure of an agent based upon its negotiation be-

[5] Numeric methods, such as *active sets*, can handle such problems [26]. However, with this method as the number of decision variables increases then so does the complexity of the computation involved in solving the quadratic problem. Therefore, active sets become unlikely candidates for computing the Nash solution for bargaining problems involving a large number of decision variables.

haviour is difficult. The most common means of doing this is to construct an explicit model of the negotiation opponent and then update and refine this model in the light of subsequent interactions (e.g., [2, 14]). However, such models are difficult and computationally demanding to construct (especially for multi-dimensional goods), they are not well suited to situations where an agent negotiates with many opponents (one model is needed per opponent), and they require numerous negotiation encounters before any great confidence can be placed on their fidelity (see section 5 for more details). An alternative approach is not to directly model the likely choice of the negotiation opponent, but rather, to try and generate a contract that is reasonably "*similar*" or "close" to the opponent's last proposal. This is a reasonable heuristic because the opponent's most recent proposal represents an outcome that is acceptable to it. Thus a proposal that is not dissimilar, also has a reasonable chance of being acceptable. In this case, the heuristic is modeling the *domain* and not the other agent. The agent can then use this domain model to induce the *possible* default preferences of the other. For example, if the seller has demanded a payment of $\pounds 20$ for a service then a client of the service can heuristically assume that the seller will prefer an offer of $\pounds 18$ to $\pounds 10$ because the former is closer, or more similar, than the latter to the seller's initial demand. Note also that the final outcome reached is a function of the initial and subsequent offer strategy. Thus, a seller starting at an offer of $\pounds 40$ should be better off. We briefly evaluate the effect of different offer strategies on the outcome of games empirically in section 4. The computational simplicity and parsimonious usage of agent models in this similarity-based approach are demonstrated in the following sections. Moreover, similarity can be applied to encounters between agents that have never previously interacted[6]. For these reasons, we use similarity as the basis for computing trade-offs in our algorithm.

3 Making Trade-Offs

This section presents a formal model of our trade-off mechanism (section 3.1), details the algorithm for actually making trade-offs (section 3.2) and illustrates its use (section 3.3). Firstly, however, we outline the basics of our service-oriented negotiation model (refer to [12] for more details). Let i ($i \in \{a, b\}$) represent the negotiating agents and j ($j \in \{1, ..., n\}$) be the decision variables under negotiation. Negotiations can range over quantitative (e.g. price, delivery time, and penalty) or qualitative (e.g. quality of service) decision variables. Quantitative decision variables are defined over a real domain (i.e. $x^i_j \in \mathcal{D}^i_j = [min^i_j, max^i_j]$). Qualitative decision variables are defined over a partially ordered set (i.e. $x^i_j \in \mathcal{D}^i_j = \{q_1, q_2, \ldots, q_p\}$). Each agent has a scoring function $V^i_j : \mathcal{D}^i_j \rightarrow [0, 1]$ that gives the score it assigns to a value of decision variable j in the range of its acceptable values. For convenience, scores are kept in the interval $[0, 1]$. The relative importance that an agent assigns to each decision variable under negotiation is modeled as a weight, w^i_j, that gives the importance of decision variable j for agent i. We assume the weights of both agents are normalized, i.e. $\sum_{1 \leq j \leq n} w^i_j = 1$, for all $i \in \{a, b\}$. An agent's scoring function for a *contract*—that is, for a value $\mathbf{x} = (x_1, ..., x_n)$ in the multi-dimensional space defined by the decision variables' value ranges is then defined as: $V^i(\mathbf{x}) = \sum_{1 \leq j \leq n} w^i_j \cdot V^i_j(x_j)$ [7].

We assume both parties have a deadline by when they must complete the negotiation. This time can be different for each agent and if its deadline passes the agent withdraws from the negotiation

[6] The similarity heuristic can be more efficient if more information about the negotiation opponent is available. Thus if the agent does have some information about its opponent's preferences, this will improve the trade-offs that are generated.

[7] For analytical purposes we restrict ourselves to an additive and monotonically increasing or decreasing value scoring system. Note that the heuristic trade-off *model* presented here is independent of the way utilities are computed. The only requirement of the model is that there exists a (linear or non-linear) utility function. However, the hillclimbing *algorithm* presented in this paper assumes agents have linear utility functions.

(taking the conflict outcome). An agent accepts a proposal when the value of the offered contract is higher than the offer it is ready to send out at that moment in time.

3.1 A Formal Model

In choosing to make a trade-off negotiation action an agent is seeking to find a contract that has the same score as its previous proposal for itself, but which may be more acceptable to (have higher score for) its negotiation opponent. However, the key problem here is how to select a contract that is likely to increase the score of the opponent, *given that the agent does not know its preferences.* To make trade-offs under these circumstances, an agent (call this a) in negotiation with another agent (call this b) must be provided with a mechanism to:

1. select a set of contracts all of which have the same utility as $a's$ previous offer $\mathbf{x}$ (this is called $a's$ aspirational level);

2. select from this set, a contract ($\mathbf{x}'$) that agent a *believes* is more preferable to b than $\mathbf{x}$. Ideally, a would like to choose the one that is most preferred by b since this maximises the chances of it being accepted.

That is, agent a believes that $V^b(\mathbf{x}') > V^b(\mathbf{x})$. By construction, the constraint $V^a(\mathbf{x}) = V^a(\mathbf{x}')$ has to be true. Thus, it follows that agent a believes that $V^a(\mathbf{x}') + V^b(\mathbf{x}') > V^a(\mathbf{x}) + V^b(\mathbf{x})$ and therefore believes that $\mathbf{x}'$ increases the joint utility of the proposal.

The first problem to address in this section is how to model the agent's uncertain belief in the second step of the mechanism's operation. A classic solution for handling such uncertainties is to assume agents have means to compute conditional probabilities and formulate subjective expected utilities. However this approach is problematic. Firstly, assigning prior probabilities is practically impossible for the types of problems addressed here (where there can be an infinitely large set of outcomes and the outcome set itself can change dynamically in the course of the negotiation). Even if assigning prior probabilities was practically achievable for interactions that are repeated (hence permitting the use of probability update mechanisms such as Bayes rule [47]), the same is not true for encounters in open systems. In such environments the prior probabilities may simply be wrong, a fact that is exacerbated by the one-off nature of encounters which prevents the update of prior distributions. Secondly, the formulation of decisions based on subjective expected utility introduces the silent out-guessing problem [63]—the agent designer's choice of probabilities is based on guesses about the probable choices of others, whose choice in turn is dependent on the guesses about the probable choices of the first, and so on.

To circumvent these difficulties a solution was sought that is simple and applicable to the types of problems that are present in both closed and open systems. As discussed previously, the heuristic employed here is not to directly model the likely choice of the other agent, but, rather, to select the contract that is most "similar" or "close" to the opponent's last proposal (since this may be more acceptable to the opponent).

The rationale for similarity-based reasoning is demonstrated by the following service selling scenario. The service provider's main negotiation objective is to sell the service. How good and how successful a service is can only be known a posteriori, once its acceptance in the marketplace can be evaluated. A common way of approximating this acceptance is to perform a poll that allows the market participants' preferences to be elucidated. However this is difficult in areas where the number of opponents is small and each is selfishly motivated to alter their answers in order to influence the service's evaluation. Therefore, statistical inference may lead us to the wrong conclusions. To circumvent this, we need an a priori valuation of the service in order to drive the negotiation process. The classical way of doing this is to organise the service's valuation around a set of characteristics that determine its differentiation from competing services. These characteristics then become the cause of the consumer's satisfaction. However, valuations based on a service's characteristics are in essence subjective, they can be wrong, the service may, in the end, not be satisfactory to the market. With this background, our research philosophy for modeling a priori valuations can be stated as: similar services should be indifferent to customers. Moreover, the greater the degree of similarity, the more likely there is to be indifference. This is also consistent with Hume's stance: "from causes which appear similar we expect similar effects. That is the sum of all our experimental conclusions". In a sense, if we accept that the a priori valuation of a good must be grounded on its characteristics, we have to accept that goods considered as similar in the light of these characteristics must receive similar valuations. Note that here the similarity function is being used to induce a utility structure (in terms of indifference structures [3]), the more similar an object the more indifferent the valuation.

The particular means of computing similarity that we adopted here is that of fuzzy similarity [64]. This shift in emphasis from the probable choices of others to the *closeness* of two contracts means that any theory that makes the same ontological commitments as classical logic and probability theory (where facts are either true or not and probabilities represent the degree of *belief*) is inappropriate. Thus, when modeling concepts such as closeness, tallness or heaviness a different logic is required that models the degree of *truth*—a sentence is "sort of" true. Most people would hesitate to say whether the sentence "Jeni is tall" is true or not, but would more likely say "sort of". Note, this is not an uncertainty about the external world (we are sure how tall Jeni is), rather it is a statement about the vagueness or uncertainty over the linguistic term "tallness" or the similarity/membership of a class prototype. However, an important point to note is that the use of fuzzy similarity and probabilities are not exclusive. Indeed, the agent can use fuzzy similarities to guess the prior probabilities of the other's choices and then update these prior probabilities in the course of interactions using Bayes rule. Thus, fuzzy similarity can be used to "bootstrap" decision mechanisms that operate on the basis of choice distributions.

We first introduce the basic concepts of fuzzy similarity and in the next section detail their usage to model trade-offs. The first thing to model is how to compute similarity along a dimension of the negotiation space (i.e., the similarity for a particular decision variable). A graded (or fuzzy) similarity relation can be seen as a generalization of an equivalence relation and it is also closely related to the mathematical notion of distance. Indeed, from the perspective of the fuzzy set literature, a fuzzy similarity relation on a set $\mathcal{D}$ is a binary function $Sim : \mathcal{D} \times \mathcal{D} \to [0, 1]$ satisfying the three following properties:

(i) *reflexivity*: $\forall x_j \in \mathcal{D}, Sim(x_j, x_j) = 1$,

(ii) *symmetry*: $\forall x_j, y_j \in \mathcal{D}, Sim(x_j, y_j) = Sim(y_j, x_j)$, and

(iii) *t-norm transitivity*: $\forall x_j, y_j, z_j \in \mathcal{D}$, If $Sim(x_j, y_j) = a$ and $Sim(y_j, z_j) = b$ then $Sim(x_j, z_j) \geq T(a, b)$, where T is a t-norm.[8]

Notice that if Sim is a similarity function in the above sense, $d = 1 - Sim$ has the properties of a distance-like function. In particular, for $T(u, v) = \max(0, u + v - 1)$ (Lukasiewicz t-norm) property

[8]A triangular norm, t-norm for short, is a binary, commutative, associative, non-decreasing operation in [0, 1] with 1 as a neutral element. T-norms play a central role in fuzzy set theory in modeling intersection operations on fuzzy sets [38].

(iii) is nothing but the usual triangular inequality and d becomes a pseudo-metric, while for $T = \min$, d becomes an ultra-metric (it verifies $d(x, y) \leq \max(d(x, z), d(z, y))$).

In this work, the method of building similarity functions is to define, for a given decision variable, criteria evaluation functions. That is, functions that determine how much, in the scale $[0, 1]$, a given element matches the criteria. For instance, in the domain of colours, a criteria evaluation function could be *temperature* that operates by returning a higher value for increasingly warm colours. Thus, given a criteria evaluation function $h : \mathcal{D} \rightarrow [0, 1]$, a natural way to define a similarity function induced by h is to define $Sim_h(x, y) = h(x) \leftrightarrow h(y)$, where $\leftrightarrow$ is a fuzzy equivalence operator, somehow related to the t-norm T to guarantee property (iii). For instance, for $T(u, v) = \max(0, u + v - 1)$, we define $h(x) \leftrightarrow h(y) = 1 - |h(x) - h(y)|$, and for $T = \min$, we define $h(x) \leftrightarrow h(y) = 1$ if $h(x) = h(y)$, and $h(x) \leftrightarrow h(y) = \min(h(x), h(y))$ otherwise.

Now, if we need to define not one, but a set of criteria functions $h_i : \mathcal{D} \rightarrow [0, 1]$ the question is how can we aggregate the individual similarities Sim_{h_i} to come up with a global similarity relation that takes into account all the given criteria? Following the results from [56], such a similarity function can always be defined as the minimum of appropriate fuzzy equivalence relations induced by a set of $m \geq 1$ criteria functions $h_i : \mathcal{D} \rightarrow [0, 1]$. That is, the similarity between two values for decision variable j, $Sim_j(x_j, y_j)$, could be defined as $Sim_j(x_j, y_j) = \min_{1 \leq i \leq m}(h_i(x_j) \leftrightarrow h_i(y_j))$. This definition, although providing a procedure to build a similarity relation from a set of criteria functions, has a very counter intuitive interpretation. If, for example, we had ten criteria functions and that for a concrete pair of elements nine of them give a high value and one of them gives a very small value, the similarity of the two elements would be equal to that minimum value. This is too strict. A better alternative, and the one that will be used in the remainder of this paper, is to build similarity functions as weighted means. By doing this, we may no longer guarantee the t-norm transitivity for the global similarity. However properties (i) and (ii) are the most important in this context and they are sufficient to model the concept of *closeness* intended in this paper. Nevertheless, t-norm transitivity is indeed preserved when the functions Sim_{h_i} are Lukasiewicz-transitive (i.e. when $1 - Sim_{h_i}$ are metrics) and $u \leftrightarrow v = 1 - |u - v|$. Thus our definition for a similarity is the following:

Definition 1 *Given a domain of values $\mathcal{D}_j$, a similarity between two values $x_j, y_j \in \mathcal{D}_j$ is defined as:*

$$Sim_j(x_j, y_j) = \sum_{1 \leq i \leq m} w_i \cdot (h_i(x_j) \leftrightarrow h_i(y_j)) \quad (1)$$

where w_i, $\sum_{1 \leq i \leq m} w_i = 1$, is a set of appropriate weights representing the importance of the criteria functions in the computation of similarity, and $1 - |h(x_j) - h(y_j)|$ is the equivalence operator (as argued before). These weights model different stances with respect to a particular decision variable. For instance, when buying a car, young people may give more importance to the luminosity of a colour because it helps in showing off, while older people may give more importance to the visibility of a colour as this is correlated with security.

To illustrate the modeling of similarity for a decision variable, consider the example of colours. Here $\mathcal{D}_{colours} = \{yellow, violet, magenta, green, cyan, red, \ldots\}$. To model how similar two given colours are, different perceptive criteria can be considered. For instance, there are 'warm' colours and 'cold' colours. With respect to this criterion, *yellow* and *orange* are more similar than *yellow* and *violet.* Related to the 'warmness' of colours, Newton [33] established the proportionality factors between colours that determine what the size of painted surfaces should be in order to be in perceptual equilibrium. For instance, yellow has luminosity 9 and violet luminosity 3. This means that if we paint two squares, one in yellow and one in violet, their surfaces have to be in relation 1 to 3 in order for the result to be in 'equilibrium' (that is, the yellow square must be one third of the size of the violet square). Another relevant perceptual criterion of colours is their visibility. There are various physiological characteristics of the human visual field, distribution of cones and rods, that ensure some colours are better perceived when moving away than others [28]. Green is the colour with the worst visibility and yellow and cyan are those with the best visibility. Given these three criteria, the colour domain can be modeled in the following way (functions are presented extensively as sets of pairs (input, output)):

$$h_t = \{(yellow, 0.9), (violet, 0.1), (magenta, 0.1), (green, 0.3), (cyan, 0.2), (red, 0.7), \ldots\}$$

$$h_l = \{(yellow, 0.9), (violet, 0.3), (magenta, 0.6), (green, 0.6), (cyan, 0.4), (red, 0.8), \ldots\}$$

$$h_v = \{(yellow, 1), (violet, 0.5), (magenta, 0.4), (green, 0.1), (cyan, 1), (red, 0.2), \ldots\}$$

where h_t, h_l and h_v are the criteria functions corresponding to temperature (warm is 1, cold is 0), luminosity (maximum is 1, minimum 0) and visibility (again maximum is 1 and minimum 0) respectively. Assume that it is a young person buying the car who has the following weights for the different criteria: $w_t = 0.7, w_l = 0.2, w_v = 0.1$. Then, using the similarity relation as defined above we have:

$$\begin{aligned}
&Sim_{colour}(yellow, red) = \\
&\quad w_t \cdot (1- \mid h_t(yellow) - h_t(red) \mid)+ \\
&\quad w_l \cdot (1- \mid h_l(yellow) - h_l(red) \mid)+ \\
&\quad w_v \cdot (1- \mid h_v(yellow) - h_v(red) \mid) \\
&\quad = 0.7 \cdot 0.8 + 0.2 \cdot 0.9 + 0.1 \cdot 0.9 = 0.83
\end{aligned}$$

and similarly, $Sim_{colour}(yellow, violet) = 0.7 \cdot 0.2 + 0.2 \cdot 0.4 + 0.1 \cdot 0.5) = 0.27$

Once the notion of similarity for a decision variable has been defined, the similarity between two contracts is simply defined as a weighted combination of the similarity of the decision variables:

Definition 2 *The similarity between two contracts* $\mathbf{x}$ *and* $\mathbf{y}$ *over the set of decision variables* J *is defined as:*

$$Sim(\mathbf{x}, \mathbf{y}) = \sum_{j \in J} w_j^a \cdot Sim_j(x_j, y_j) \tag{2}$$

with $\sum_{j \in J} w_j^a = 1$ and Sim_j being the similarity function for decision variable j defined as before. These weights represent the level of importance the agent believes the opponent places on the different decision variables. If an agent has no such information, it may assign equal weights to all decision variables. However if it can deduce the likely priorities of its opponent, then these weights can be modified to reflect this information.

Given this background, we can now proceed with the details of the trade-off formal model. An agent will decide to make a trade-off action when it does not wish to decrease its aspiration level (denoted θ) for a given service-oriented negotiation (the aspiration level is the valuation of its last offer $\mathbf{x}$, that is $\theta = V(\mathbf{x})$). Thus, the agent first needs to generate some/all of the potential contracts for which it receives the score of θ. Technically, it needs to generate contracts that lie on the iso-value (or indifference) curve for θ [41]. Because all these potential contracts have the same value for the agent making the trade-off, it is indifferent amongst them. Given this fact, the aim of the trade-off mechanism is to find the contract on this line that is most preferable (and hence acceptable) to the negotiation opponent. More formally, an iso-curve is defined as:

Definition 3 *Given an aspirational scoring value θ, the iso-curve at level θ for agent a is defined as:*

$$iso_a(\theta) = \{\mathbf{x} \mid V^a(\mathbf{x}) = \theta\} \tag{3}$$

From this set, the agent needs to select the contract that is most similar to agent b's last offer. A trade-off is then defined as:

Definition 4 *Given an offer, $\mathbf{x}$, from agent a to b, and a subsequent counter offer, $\mathbf{y}$, from b to a, with $\theta = V^a(\mathbf{x})$, a trade-off for agent a with respect to $\mathbf{y}$ is defined as:*

$$\textit{trade-off}_a(\mathbf{x}, \mathbf{y}) = \arg \max_{\mathbf{z} \in iso_a(\theta)} \{Sim(\mathbf{z}, \mathbf{y})\} \tag{4}$$

A linear trade-off algorithm that implements an *instance* of this generic formal heuristic model is described next.

3.2 The Trade-off Algorithm

The trade-off algorithm we consider here is defined over the class of linearly additive utility functions. We acknowledge that restriction to a linear utility model limits the applicability of the algorithm. However, we also note that the assumption of linearity is restricted to the algorithm and not the heuristic model itself. It is perfectly consistent with the heuristic model to design other trade-off algorithms for other non-linear utility functions (see [10] for non-linear distributed search algorithms).

This algorithm performs an iterated hill-climbing search in a landscape of possible contracts. The search starts at the opponent's offered contract and proceeds by generating a set of contracts that lie closer to the iso-curve (representing the agent's aspiration level). The contract that maximizes the similarity to the opponent's last offering is selected at the end of each iteration. The algorithm repeats, starting from the contract selected at the previous step, until the iso-curve is eventually reached.

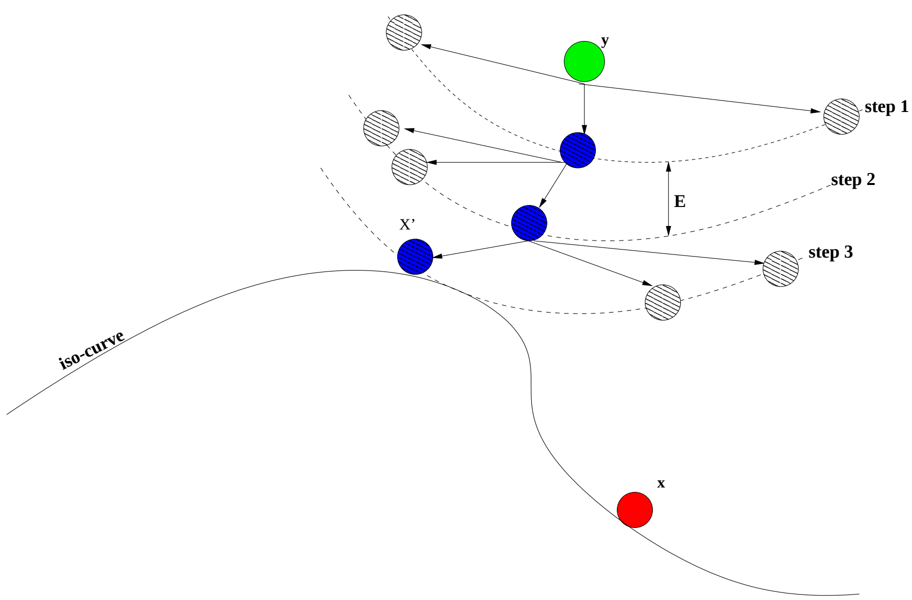

Figure 3: Schema of the trade-off algorithm with $N = 3$ and $S = 3$.

The algorithm is shown schematically in figure 3. It starts at contract $\mathbf{y}$, the opponent's last offer, and moves towards the iso-curve (the solid marked line corresponding to the agent's aspiration level θ) associated with $\mathbf{x}$, the agent's last offer. This approach to the iso-curve is performed sequentially in S steps (three in figure 3). Each step starts by randomly generating N contracts (three, one filled and two patterned ovals in figure 3) that have a utility E greater than the contract selected in the last step $\mathbf{y^j}$ (or $\mathbf{y^0} = \mathbf{y}$ if it is the first step) for the agent making the trade-off. Here N is referred to as the number of children. Each new contract $\mathbf{y^{j+1}}$ so generated satisfies the constraint $V(\mathbf{y^{j+1}}) = V(\mathbf{y^j}) + E$, and they all have the same utility to the agent making the trade-off (shown as the dotted line connecting all the children at each step). From the generated child contracts, the one that maximizes the similarity with respect to the opponent's contract $\mathbf{y}$ is selected (shown as the filled oval in figure 3). This contract then becomes the parent of the next set of children. E is computed as the overall difference between the value of $\mathbf{x}$ and $\mathbf{y}$ divided by the number of steps. That is, $E = \frac{V(\mathbf{x}) - V(\mathbf{y})}{S}$. The overall effect of the algorithm is to sequentially explore a subset of the possible space of contracts and select for the next step the one that maximizes the similarity with respect to the other agent's contract offer. This search terminates when a contract $\mathbf{x'}$ is generated that lies on the iso-curve of $\mathbf{x}$.

Figure 4 presents the part of the algorithm responsible for generating a new trade-off contract. This algorithm will thus be invoked N times at each step in order to compute the best trade-off contract (giving SN calls in total). The algorithm generates children by splitting the step gain in utility, E, randomly among the set of decision variables under negotiation.

The algorithm shows only the computations involved in making a single step, of size E, towards the iso-curve specified by $\mathbf{x}$. It functions as follows. Firstly, the maximum utility that can be gained for each decision variable, either qualitative or quantitative, is computed as the difference between the full aspiration of the agent's preferences and the utility of the decision variable's value in the contract that is being modified $V^i(y_i^j)$ (line 1). Note, at the first step of the algorithm's iteration, $\mathbf{y^0}$ will be the opponent's offered contract. Each *weighted* individual utility gain is then summed to determine the overall weighted amount of utility that can be gained (line 2). Next, because the "consumption" of this utility gain has a random element (line 5), a degree of tolerance is included to guarantee the convergence of the algorithm[9] (line 3). The process of consumption of the total available utility (computed in line 2) begins by allowing each decision variable to consume a random amount (line 5) within the limits of the interval computed in line 1 for the quantitative decision variables or by randomly selecting one of the possible finite increments for qualitative decision variables. The store of the current total amount consumed E_n is then updated as the addition of the old store and a linear weighted sum of each of the individually consumed utilities (line 6). The total amount that can be consumed is then recomputed given the newly consumed amount (line 7). If the amount consumed is less than the total amount E, the process of consumption continues until the maximum (E or the step size in figure 3) is reached. Finally, the utility gained by each decision variable is remapped to actual values that correspond to the new utility (line 9). In the case of qualitative decision variables $V_i^{-1}(u)$ must be interpreted as a function that selects a qualitative value $q \in \mathcal{D}_i$ that satisfies $V_i(q) = u$. Given that we assume a partial order we may have more than one value q with valuation u. If this is the case, we chose one randomly. The algorithm guarantees by construction that there is at least one qualitative value with valuation u.

A theoretical analysis shows that the average time the algorithm takes to complete is linear with respect to the number of decision variables in the negotiation (see [11] for details of the proof). This

[9]As the convergence is asymptotic to the value $V(y) + E_{max}$, if we had a situation with $E_{max} = E$ we could not guarantee reaching the iso-curve. Also, the search process reaches the iso-curve within epsilon distance if there is at least one decision variable over a continuous domain. Price at least plays this role in service-oriented negotiation domains.

inputs: y^j; /* last step best contract. $y^0 = y$ */
E; /* step utility increase */
$V_i()$; /* value scoring functions for the decision variables */
w_i; /* importance weights for the decision variables */
Output: y^{j+1}; /* child of y^j */
begin
for each decision variable i **do**
if i is discrete
(1) **then** $\overline{E}_i := \{\Delta_u(q) | q \in \mathcal{D}_i, \Delta_u(q) = V_i(q) - V_i(y_i^j) > 0\}$
(1) **else** $\overline{E}_i := [0, 1 - V_i(y_i^j)]$
endfor;
(2) $E_{max} := \sum_i w_i \cdot \max(\overline{E}_i)$;
(3) $\delta := 0.01 \cdot E_{max}$;
if $(E_{max} > E + \delta)$ **then**
begin
(4) $k := 0; E_n := 0$;
while $(E_n < E)$ **do**
$k := k + 1$;
for each decision variable i **do**
if $(E_n < E)$
then if i is qualitative
(5) **then** $r_i^k := random(\{\Delta_u(q) | \Delta_u(q) \in \overline{E}_i, \Delta_u(q) \leq \frac{E - E_n}{w_i}\} \cup \{0\})$
(5) **else** $r_i^k := \min(random(\overline{E}_i), \frac{E - E_n}{w_i})$
else $r_i^k := 0$;
(6) $E_n := E_n + w_i \cdot r_i^k$;
if i is qualitative
(7) **then** $\overline{E}_i := \{\Delta_u(q) | q \in \mathcal{D}_i, \Delta_u(q) = V_i(q) - \left(V_i(y_i^j) + \sum_{i \leq j \leq k} r_i^j\right) > 0\}$
(7) **else** $\overline{E}_i := [0, \max(\overline{E}_i) - r_i^k]$
endfor
endwhile;
for each decision variable i **do**
(8) $E_i := \sum_{j=1}^{k} r_i^j$;
(9) $y_i^{j+1} := V_i^{-1}\left(V_i(y_i^j) + E_i\right)$
endfor
end
else $raise\ error\ no\ step\ can\ be\ performed$
end

Figure 4: Contract generation part of the trade-off algorithm

linearity is a highly desirable property given the aim of this research to develop decision mechanisms that respect an agent's computational limitations.

3.3 A Trade-Off Scenario

To illustrate our model consider the example of a car-dealer (of name b) negotiating the purchase of a car. Assume agent a enters the garage and receives the initial proposal $\mathbf{x} = (green,\ £27000, 10weeks)$ for a deal on buying a car of a given model (over $decision\ variables = \{colour, price, delivery\}$). Clearly, the first decision variable is a qualitative one with the same domain as the colour example introduced before, and the other two are quantitative. Agent a responds to this proposal with a counterproposal $\mathbf{y} = (yellow,\ £21000, 0weeks)$ The point now is what could be a potential answer from the dealer using our trade-off technique? To answer this, we have to specify domains, weights, valuation functions and the similarity function for the car dealer:

$$\begin{array}{ll} \mathcal{D}^b_{colour} & = \{yellow, violet, magenta, green, cyan, red\} \\ \mathcal{D}^b_{price} & = [£18000,\ £35000] \\ \mathcal{D}^b_{delivery} & = [0weeks, 16weeks] \end{array}$$

We assume the following valuation functions (V^b_{colour} is extensionally defined, and the other two are linear functions):

$$\begin{array}{ll} V^b_{colour} & = \{(yellow, 0.5), (violet, 0.2), (magenta, 0.3), \\ & \quad (green, 0.8), (cyan, 0.3), (red, 0.8)\} \\ V^b_{price}(x_{price}) & = \frac{x_{price}-18000}{35000-18000} \\ V^b_{delivery}(x_{delivery}) & = \frac{x_{delivery}}{16} \end{array}$$

Finally, we assume the following weights: $w_{colour} = 0.1$, $w_{price} = 0.8$, $w_{delivery} = 0.1$.

Similarity for price and delivery will each be based on a single criteria: 'low price' (lp) and 'low delivery' (ld) respectively. These will also be modelled as linear functions:

$$h_{lp}(x) = \begin{cases} 1 - \frac{x}{40000} & x \in [0, 40000] \\ 0 & \text{otherwise} \end{cases} \qquad h_{ld}(x) = \begin{cases} 1 - \frac{x}{28} & x \in [0, 28] \\ 0 & \text{otherwise} \end{cases}$$

With all these elements, we can exemplify the working of the algorithm. First of all, from the car dealer's perspective, contracts $\mathbf{x}$ and $\mathbf{y}$ have different values: $V^b(\mathbf{x}) = 0.1 \cdot 0.8 + 0.8 \cdot \frac{27-18}{35-18} + 0.1 \cdot \frac{10}{16} = 0.558$. This value represents the car dealer's aspiration level θ. The value of agent's a offer is $V^b(\mathbf{y}) = 0.19$. Now if we run the algorithm for one step, $S = 1$, and three children per step, $N = 3$, it could generate the following trade-offs:

$$\mathbf{x_1} = (yellow,\ £28132, 5weeks), \mathbf{x_2} = (red,\ £26568, 12weeks), \mathbf{x_3} = (violet,\ £28506, 7weeks)$$

All of them verify, by construction and because we are running the algorithm for just one step, that $V^b(\mathbf{x_1}) = V^b(\mathbf{x_2}) = V^b(\mathbf{x_3}) = \theta$. Now, the trade-off algorithm selects the one with highest similarity with respect to the offer made by agent a, that is contract $\mathbf{y}$, using the car dealer's decision variable weights.

$$Sim(\mathbf{y}, \mathbf{x_1}) = 0.1 \cdot Sim_{colour}(yellow, yellow) + 0.8 \cdot Sim_{price}(£21000,\ £28132) + 0.1 \cdot Sim_{delivery}(0weeks, 5weeks) = 0.1 \cdot 1 + 0.8 \cdot 0.821 + 0.1 \cdot 0.82 = \mathbf{0.839}$$

$$Sim(\mathbf{y}, \mathbf{x_2}) = 0.1 \cdot Sim_{colour}(yellow, red) + 0.8 \cdot Sim_{price}(£21000,\ £26568) + 0.1 \cdot Sim_{delivery}(0weeks, 12weeks) = 0.1 \cdot 0.83 + 0.8 \cdot 0.861 + 0.1 \cdot 0.571 = \mathbf{0.828}$$

$$Sim(\mathbf{y}, \mathbf{x_3}) = 0.1 \cdot Sim_{colour}(yellow, violet) + 0.8 \cdot Sim_{price}(£21000,\ £26568) + 0.1 \cdot Sim_{delivery}(0weeks, 7weeks) = 0.1 \cdot 0.27 + 0.8 \cdot 0.812 + 0.1 \cdot 0.75 = \mathbf{0.751}$$

Given these values, the algorithm would chose $\mathbf{x_1}$ as the trade-off to offer to customer a. That is, $\mathbf{x}' = (yellow,\ £28132, 5weeks)$

4 Experimental Analysis

A series of experimental tests have been undertaken to calibrate the operational performance of our trade-off algorithm. Two types of empirical information were sought [10]. The first set, here referred to as single-offer experiments (section 4.2), aimed to investigate the *parameters* of the trade-off algorithm in the generation of a *single* offer (i.e., they evaluated the kernel of the algorithm). Conversely, the aim of the second set, here referred to as meta strategy experiments (section 4.3), was to investigate the *process* of negotiation when agents use trade-off and/or responsive negotiation mechanisms (i.e., they deal with the dynamics of the algorithm when interacting with other mechanisms). Recall that the latter mechanism implements an iterated search for a contract with a value that is acceptable to both parties.

4.1 Experimental Procedures

Both types of experiment involve offers from one negotiator, a $player$, to another, the $opponent$. Furthermore, both experiments involve negotiation over four quantitative decision variables $[price$, $quality, time$, $penalty]$. The domains of values of each decision variable for both agents are the same. The importance weight vectors of the agents (section 3) are fixed throughout the negotiation: $W^{player} = [0.1, 0.5, 0.25, 0.15]$ and $W^{opponent} = [0.5, 0.1, 0.05, 0.35]$[11]. The value function V_i^a used by agent a for decision variable i is a linear scoring function of the following type:

$$V_i^a(x_i) = \begin{cases} \frac{max_i^a - x_i}{max_i^a - min_i^a} & \text{if decreasing} \\ \frac{x_i - min_i^a}{max_i^a - min_i^a} & \text{if increasing} \end{cases}$$

where increasing and decreasing refer to the direction of change in score as the value of that decision variable increases. For example, increasing the $price$ of the service usually decreases the score for a client, but increases it for a seller.

The other input variables of the trade-off algorithm were set as follows. The discriminatory power—the magnitude of the difference between the input and output—of the criteria functions (equation 1) were set so that they exhibited two properties. Firstly, they have more discrimination within the decision variables' reservation values (as compared to values outside this range), since most of the negotiation will take place in this region. Thus, maximal discrimination should be between a decision

[10]The results shown are first case approximations, derived for single case rather than long term expected performance of the algorithm.

[11]Generally speaking, the differences in these weights are one of the key elements that provide the opportunity for joint improvements, the other being the different shapes of the negotiators' scoring functions (recall the discussion of section 2). For example, an increase in $price$ may have little effect in value for the $player$, but relatively more for the $opponent$.

variable's min and max values (section 3). For example, consider a buyer of a good with a single decision variable *quantity* of the good needed which has $[10, 20]$ minimal and maximal values respectively. Given this reservation, we want the criteria function to return a full ordering of values within this interval and equivalent orderings exterior to this interval. We parameterised this reservation value requirement by the independent variable ϵ. When ϵ is low, the function should be maximally discriminative for values within the decision variable's reservation limits (*mutatis mutandis* when ϵ is high). Secondly, we also want to experiment with different discriminatory powers *within* the reservation range (to support different similarity measures for different decision variables). For example, for one decision variable it may be desirable to have maximal discrimination at the center of the reservation values (e.g. within the sub ranges of $[14, 16]$ for the quantity of the overall $[10, 20]$ reservation for the quantity example given above), whereas for another decision variable maximal discrimination may be desired at the extremes of the reservation values (e.g. within the sub ranges of $[18, 20]$ for the quantity of the overall $[10, 20]$ reservation for the quantity example given above). We parameterise this requirement using the variable α. When α is high, more discrimination is placed towards the maximum of the reservation values (*mutatis mutandis* when it is low). Given this, the following function satisfies these two requirements:

$$h(x) = \frac{1}{\pi} atan \left[\left(\frac{2|x - min|}{x - min} \cdot \left| \frac{x - min}{max - min} \right|^{\alpha} - 1 \right) \; tan(\pi(\frac{1}{2} - \epsilon)) \right] + \frac{\pi}{2}$$

In this case, in order to be reasonably discriminatory, ϵ was fixed at 0.1 for all decision variables. For all decision variables, we fixed the different αs to be equal, $\alpha^{price} = \alpha^{quality} = \alpha^{time} = \alpha^{penalty} = 1$, to have linear criteria functions that have equal discrimination power across the decision variable's reservation values. We chose to make ϵ and α constant to reduce the number of free variables in the experiments (normally they would be set to reflect the agent's knowledge of a given domain).

4.2 Single-Offer Experiments

In these experiments the independent variables were: i) the number of children generated at each step in hillclimbing to the iso-curve (N in section 3.2); ii) the number of steps taken to reach the iso-curve (S in section 3.2); iii) the information that is available to an agent regarding the importance the opponent places on each decision variable in computing the contract's value (the weights in equation 2); and (iv) the $opponent's$ and the $player's$ last offers ($\mathbf{x}$ and $\mathbf{y}$ in equation 4). Values for the first and second variables control the amount of search performed by the algorithm. Experiments were run where the number of children was selected from the set $\{5, 100, 200\}$. The number of steps to the iso-curve was selected from the set $\{1, 40\}$. The specific numbers for both N and S signify very little; the important thing is the relative relationship between them. Thus, more computation is involved when the algorithm generates 200 rather than 5 children at each iteration, or when it takes 40 steps rather than 1 to reach the iso-curve. For the third set of independent variables, an agent can have perfect, partial, imperfect or uncertain information on how the other agent weights the decision variables that are input into its similarity function. In experiments with perfect information, the algorithm, in computing similarity, is given the other agent's precise weights for different decision variables (cardinally correct information). Partial information games are where the algorithm is given the correct order of importance but not the actual decision variable weights (ordinally correct information). Imperfect games represent the situation where the weight of each decision variable of the other agent is selected from a normal distribution. Finally, uncertain information games represent cases where the algorithm is given undifferentiated weights for each decision variable (in this case $[0.25, 0.25, 0.25, 0.25]$).

The experimental procedure consisted of inputting two contracts, representing $\mathbf{x}$ and $\mathbf{y}$, into the algorithm under each of the dependent variable environments and observing the execution trace of the algorithm for *an* offer from the $player$ to the $opponent$. All input contracts ($\mathbf{x}$ and $\mathbf{y}$) were subject to the general constraint that $V^{player}(\mathbf{y}) < V^{player}(\mathbf{x})$ and $V^{opponent}(\mathbf{x}) < V^{opponent}(\mathbf{y})$. This ensured trade-offs are possible by ruling out all those contracts that are already of a higher value to either party. The control set was generated by choosing the preferred child randomly at each step approaching the iso-curve (as opposed to using the similarity criteria).

The hypotheses of these experiments are given in terms of the input and output of the trade-off algorithm. The input is the set of importance weights of the other agent (perfect, partial, imperfect and random) and the output is a contract that has the same score to the player, but some other score to the opponent. Specifically, the hypotheses are:

> **Hypothesis 1:** *The greater the exploration of the space of possible deals, the better the output of the algorithm from the perspective of the negotiation opponent.*
>
> **Hypothesis 2:** *The quality of the algorithm's output (the score of the contract to the $opponent$) is directly correlated to the quality of information input—the better the input information, the better the outcome quality.*

These hypotheses simply state the intuition that a more refined search of the possible space of contracts should result in selecting and offering a contract that has more value to the other agent. Furthermore, this search should be directly affected by the information the algorithm has about the other agent's decision variable importance rankings.

Figure 5 and the top row of figure 6 show the results of varying, under different information inputs, the number of children generated when the number of steps to the iso-curve is set to 40. The bottom row of figure 6 represents the case where the number of children is set to 100, but the trade-off algorithm computes the iso-contract in a single step. The dot-dash line represents the execution trace of the random control, the solid line emanating from $\mathbf{y}$ the similarity based trade-off execution trace, and the line joining $(0, 1)$ to $(1, 0)$ the pareto-optimal line. The pareto-optimal line was computed using the weighted method [41, 7]. The output of the algorithm, $\mathbf{x}'$, is shown in figures 5 and 6 (top row) as the end point of the execution trace and for 6 (bottom row) as the explicitly marked points (since there is no trace). For benchmarking purposes, the reference point (and not the Nash bargaining solution for reasons given in section 2), is also plotted in all cases. Note however, that the aim here is simply to observe the amount of benefit the other party gains as a function of the algorithm's performance under different contexts, rather than maximisation of any of the explicit solution concepts introduced in section 2.

Three major patterns are observed that directly and indirectly support our hypotheses. Direct support for hypothesis 1 is given by the observation that when moving to the iso-curve if the space of possible contracts is not explored sufficiently—5 children (figure 5 top row) or 1 step (figure 6 bottom row)—then the gains of the $opponent$ are at best insignificant and at worst negative. More specifically, only when the $player$ has perfect information about the $opponent$'s evaluations and the trade-off mechanism operates in 1 step with 100 children will the mechanism improve the offer (from the $opponent$'s perspective) (figure 6 E). The next best contract for the $opponent$ is when it has the same value as $\mathbf{x}$ (figure 5 A). All other contracts generated by the $player$ when it does not explore the search space (figures 5 B,C,D and 6 F) have lower value to the $opponent$ than the original offer.

However, the $opponent$'s benefit increases as the algorithm performs more search (from 5 to 200 children in 40 steps—figure 5 top row [5 children], bottom row [100 children], and figure 6 top row [200 children]). Thus, generating more children does indeed increase the utility of the opponent.

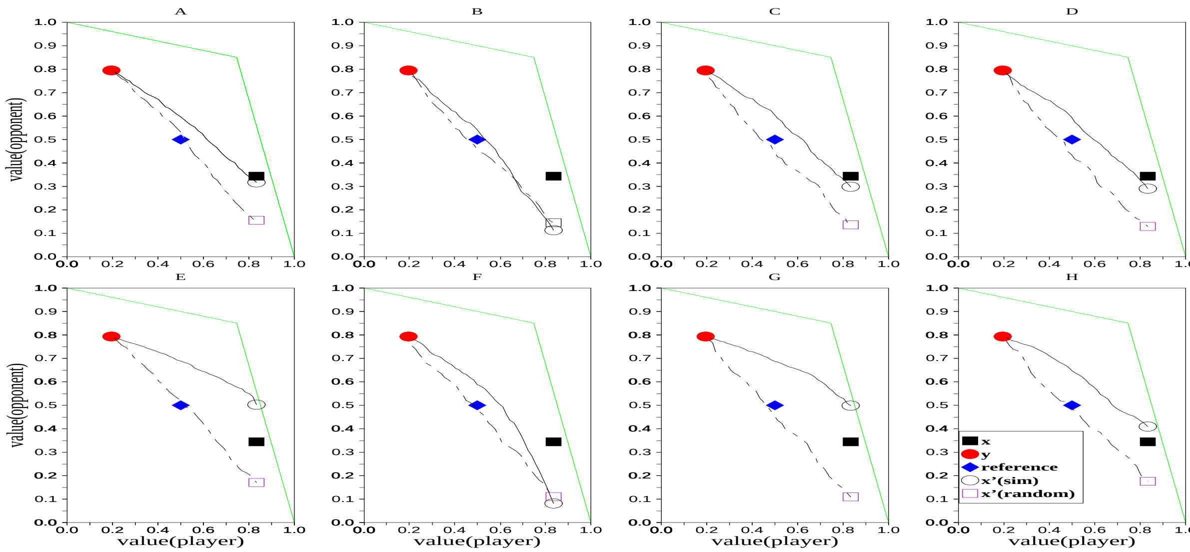

Figure 5: Data for 5 children in 40 steps (first row) and 100 children in 40 steps (second row). A) & E) perfect information, B) & F) imperfect information, C) & G) partial information, D) & H) uncertain information.

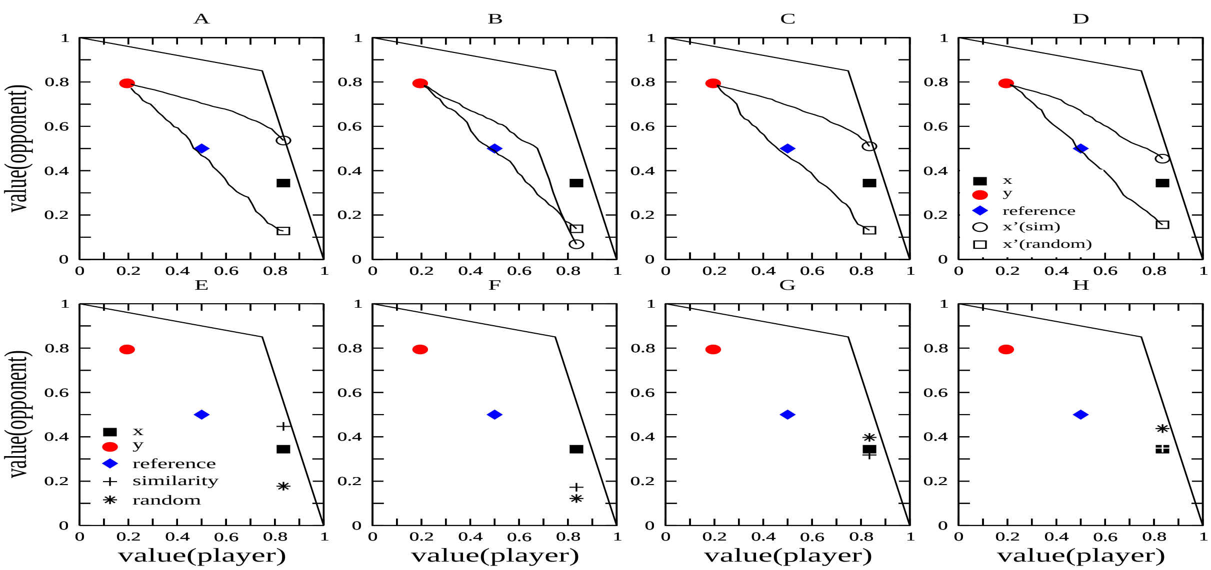

Figure 6: Data for 200 children in 40 steps (first row) and 100 children in 1 step (second row). A) & E) perfect information B) & F) imperfect information, C) & G) partial information, D) & H) uncertain information.

However, the data suggests there is a point above which generation of more children does not increase the utility of the opponent. This is observed in the lack of any significant difference between perfect and partial information outcomes within either the 100 and 200 children (40 steps) result categories (compare figures 5 E, F, G and H with 6 A, B, C and D). Furthermore, the expectation, as stated by hypothesis 2, that the more accurate the information about the weights of the $opponent$ are, the better the contract score for the $opponent$, is supported by the observation that the utility to the $opponent$ is indeed increased when the algorithm is increasingly supplied with more correct information about the $opponent's$ weights (seen as increasing utility) from the incomplete to uncertain information classes. However, the hypothesis is rebutted for perfect and partial information cases (compare figure 5 E with G or figure 6 A with C). This lack of significant differences between contracts selected under perfect and partial information conditions indicates that the algorithm requires only partial ordering information, rather than perfectly cardinal orderings, in order to compute outcomes that are better for the $opponent$. This is because the absolute differences in magnitude between the perfect and partial information classes are small, resulting in input variables that are not significantly different. The chosen value for the partial weight estimation cannot be made significantly different from the perfect weight estimation values because the actual values of the partial estimates are constrained both at the upper and lower limits by the perfect and uncertain weight estimation values.

Positive support about the relationship between the quality of the input and the resultant output is given in the final observation that, for all environments and variable combinations, imperfect information (figure 5 B and F, and figure 6 B and F) results in significantly poorer outcomes for the $opponent$ than all the other information classes. This is only to be expected since the search is directed towards erroneous directions when the information supplied about the other agent is incorrect.

Note, in nearly all cases, the similarity based trade-off out performs the policy of randomly selecting a child for the next step towards the iso-curve. However this pattern does not hold for the cases of reaching the iso-curve in one step under partial and uncertain information environments (figure 6 G and H). Given an offer is generated in 1 step, this is due to chance, rather than randomness being a better strategy in this type of environment (supported by the consistently poor performance of the random selection strategy in the experiments where the number of steps to the iso-curve is set to 40, figure 5 C, D, G and H, and 6 C and D).

In summary, these results indicate that unless agents know, at least partially, the importance the other agent attaches to a decision variable, then the best policy for computing trade-offs is to assign uncertain weightings to all decision variables. These weightings can then be updated by some learning rule towards partial or perfect information models, since a) information models are private and b) erroneous predictions can result in poorer outcomes. Furthermore, engaging in trade-off negotiation, particularly with a high search factor by both parties, results in higher joint gains.

4.3 Meta Strategy Experiments

The aim of these experiments is to empirically evaluate the outcome and dynamics of negotiation when agents use either a trade-off mechanism or a responsive mechanism or a combination of the two in the course of negotiation (that is, a meta strategy of which mechanism to select in order to generate a series of counter-proposals). The first offer of both agents was generated using responsive mechanisms, since the trade-off mechanism requires at least one offer from the opponent. After that, an agent is faced with a choice of which mechanism to select. Since the number of meta strategies is exponential on the length of the negotiation (there are as many as there are potential sequences of choosing between responsive and trade-off types of counter-proposals), the meta strategies considered here were limited to the set $\{responsive, smart, serial, random\}$. Responsive simply selected the responsive mechanism for generating an offer throughout negotiation. This was included to compare the trade-off mechanism against an agent that always concedes utility. A smart strategy consisted of deploying a trade-off mechanism until the agent observed a deadlock in the average closeness of offers between both agents as measured by the similarity function. That is, the distance between the offers was not reducing. Under these circumstances, the value of the previously offered contract, $V^a(\mathbf{x})$, was reduced by a predetermined amount, here 0.05, thereby lowering the input value of θ into the trade-off mechanism. A serial strategy involves alternating between the trade-off and responsive mechanisms. Finally, the random meta strategy randomly selected between the two mechanisms. The parameters of the responsive mechanism were set to produce concessionary behaviours, since being responsive often involves concessions in the light of environmental needs (e.g. time, resources etc.). For the trade-off algorithm, the number of children and number of steps were set to 100 and 40 respectively and the similarity weights were set at the uncertain settings of $[0.25, 0.25, 0.25, 0.25]$. Both negotiators were given a deadline of twenty offers.

The particular hypotheses we sought to evaluate here are as follows:

> **Hypothesis 3:** *The more the space of possible deals is explored* jointly*, the better the joint outcome.*
>
> **Hypothesis 4:** *Higher joint utilities are obtained at the expense of greater communication between the agents.*

These hypotheses essentially state the expectation that a symmetric game consisting of a pair of smart meta-strategies should select final outcomes that have a higher joint value than other types of meta-strategies. This is expected because a smart meta-strategy is essentially a trade-off strategy that *only* concedes a small amount when a deadlock is detected. All other experimental meta-strategies have an element of concession involved in them (since the variables of the responsive mechanism have been chosen to behave in a concessionary fashion). Thus any meta-strategy that selects a responsive mechanism in the course of negotiation (all pairs of meta-strategies except [smart,smart]) should result in joint utility execution traces that "move" south westerly, away from the pareto-optimal line. Furthermore, meta-strategies that engage more in search for higher joint utilities and less on concessions should result in higher communication loads. This latter expectation is based on the intuition that a responsive mechanism generates contracts that successively approach the point of cross over in offers faster than the trade-off mechanism. Hence it is to be expected that a meta-strategy that selects the responsive mechanism should reach deals quicker than one that is smart.

Figure 7 presents the data for the meta-strategy experiments investigating the process of mechanism selection. Individual offers between the $player$ and the $opponent$ are depicted as circles and squares respectively. The sequences of offers are joined by a solid line for the $player$ and a dotted line for the $opponent$. The final agreement is depicted as the offer where the circle and square meet. The communication load is simply the addition of the numbers of circles and the squares.

The observed rank ordering across meta-strategy pairings over the summed joint utility gained for the final outcome directly supports hypothesis 3. The highest joint gain is achieved in negotiations between two $smart$ meta-strategies. Furthermore, in this case the final outcome is closest to the axiomatic reference outcome (the pareto point that connects the reference outcome with utopia—section 2) than any other meta-strategy pairing, implying that such a pairing results in outcomes that are most beneficial to both parties. This result suggests that if agents are motivated by maximising the joint utility of the outcome then rational agents have an incentive to be symmetrically implemeting the trade-off algorithm. The remaining summed utility rankings for $player, opponent$ pairings of meta-strategies are then [smart,serial], [serial,serial], [smart,random], [smart,responsive], [serial,responsive], [ran-

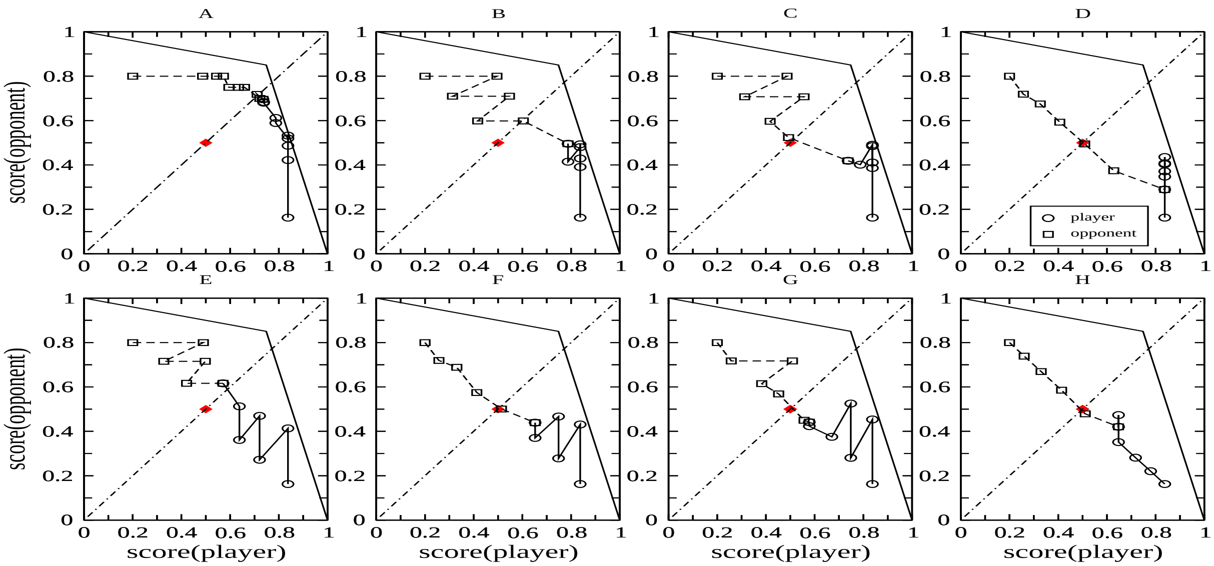

Figure 7: Dynamics of negotiation process for meta strategies: A) smart v. smart, B) smart v. serial, C) smart v. random D) smart v. responsive, E) serial v. serial, F) serial v. responsive, G) random v. random, H) random v. responsive.

dom,responsive], [random,random] with respective joint gains of $1.27, 1.18, 1.146, 1.11, 1.076, 1.06, 0.99$. In general, the higher joint utilities occur when at least one of the agents is $smart$. The $random$ meta strategists, as expected, perform worst.

Hypothesis 4 is supported by the observation of the number of messages exchanged between agents using different meta-strategies (recall that the communication load is simply the addition of the individual messages exchanged in figure 7). As predicted, the observed pattern is almost the reverse for the joint value outcomes above; with a [smart,smart] pairing incurring the highest communication cost (reaching a deal after 19 rounds (20 was the deadline)), followed by [random,random], [smart,responsive], [smart,random], [smart,serial] (14 rounds), [serial,serial] (13 rounds), and [serial,responsive] (12 rounds). This observation supports the intuition that higher joint utilities are gained through greater search, which, in turn, involves more communication between the agents.

5 Related Work

The problem of negotiation is extensive, at both the local and the social level, with subproblems that occur not only during the negotiation period itself (the gaming problem), but also at the pre and post negotiation phases (the knowledge and commitment problems respectively). Consequently, there has been a considerable body of work from different fields, ranging from operational research, management sciences, decision theory, game theory and to, more recently, autonomous computational systems. Negotiation in operational research is viewed as an optimization problem solved through the design of (mostly centralized) optimal solution algorithms [7, 17, 23, 54]. These algorithms, based on mathematical programming techniques, are often optimal because: a) the geometry of the solution set is assumed to be described by a closed and convex set (therefore there is a bounded number of solution points), b) the objective functions of the individuals (the utility functions) are concave and differentiable and c) some global information (such as the utility gradient increase vector [7]) is stored or elicited by a centralized mediator that acts to direct problem solvers towards the pareto-optimal line. However, although analytically elegant, such optimality cannot be guaranteed in decentralized autonomous agent systems operating in open environments where information is sparse and there is a lack of trust.

This algorithmic approach contrasts with the cooperative (axiomatic) and non-cooperative approaches of game theory that have been highly influential in the mechanism design tradition of MAS [23, 24, 43, 48, 58]. Rosenchein and Zlotkin used cooperative game theory to design negotiation mechanisms that maximize the social welfare function (the product of agent utilities, or the Nash solution) for task, state and worth oriented domains [43]. Similarly, Sandholm, in addition to extending the Contract Net protocol [52] with decision theoretic mechanisms, developed a computational model of leveled commitments and coalition formation based on principles of cooperative game theory. On the other hand, Kraus developed negotiation mechanisms based on non-cooperative (or strategic) game theoretic models (in particular that of Rubinstein [45], which has been shown to implement the Nash bargaining solution under some conditions [46] and thus strengthening the support for the Nash Program [31]) that models the negotiation process as a bi-lateral bargaining game, consisting of an alternating and sequential protocol of offers and counter-offers.

Our work also borrows from game theory. In particular, we adopt the nomenclature and concepts of game theory (in terms of utility maximizing agents and pareto-optimality) for developing and evaluating our negotiation mechanism. However, despite this influence, our negotiation mechanism is based on a different set of assumptions (see [8] for a critique of the various game theoretic approaches). In general, although analytically well formed [4], game theory's rationality assumption, shared by the majority of its computational extensions—that i) beliefs are common knowledge (in its strong form and probabilistically inferred in its weaker form), and ii) individuals are optimizers and computationally unbounded—is inappropriate for open system problems. These assumptions are based on an "ideal" world in which beliefs deduced rationally from a common prior can be common knowledge and computation is unbounded. However the real world is not ideal. There are imperfections in an agent's knowledge and optimization behaviour is often not independent of actual capabilities and limits. In its strongest form, the combination of the two assumptions implies that no computation is required to find mutually acceptable solutions within the feasible range. This space of possible deals is assumed to be fully known by the agents, as are the potential outcome values. Agreements are thus instantaneous. Inefficiencies only arise when beliefs are probabilistically inferred, leading to a process of negotiation. Generally, the theory is silent with respect to the actual computational rationality of the agents [51]. However in the real world, to know a solution *exists* is not to know what the solution *is*. The perfect rationality of all agents, although useful in designing, predicting and proving properties of a system, is therefore not altogether useful in system design since physical mechanisms do take time to process information and select actions. Therefore, what is required are different agent architectures that implement different search mechanisms, capable of heuristically exploring the set of possible outcomes, under both limited information and computation assumptions. In fact, heuristics were also proposed by Nash as a method of narrowing down the set of possible equilbrium strategies of a non-cooperative game [31]. In the environments in which our model operates, where agents must deliberate over an n-dimensional space of deals, rather than simpler games of dividing the dollar, solution quality is based on a satisficing rather than optimal criteria.

Uncertainty in negotiation was also addressed by using decision theoretic models in the *Persuader* system [53] where multi-attribute utility theory was combined with case-based reasoning in contexts where the agent had no previous cases to reason with. This dual approach is similar to our work in that agents use both utility and similarity for decision making. However, we use similarity rather than utility to address the inherent uncertainties involved and, as we have shown in section 4.2, this appears to be a better choice in uncertain environments.

The process of negotiation has also been modeled as a distributed constraint satisfaction problem [1, 49, 62]. In the work of Sathi and Fox, agents' objectives are represented as constraints together with their associated utilities. Strategies (e.g. composition, reconfiguration and relaxation operators) are then used to modify these constraints, or the current solution, until a final solution is reached. The relaxation of constraints is similar to our work on concession mechanisms, and the modification of the current solution closely resembles the trade-off mechanism reported here. However, in our work there is only one objective, namely reaching a contract which maximises value. Therefore, our approach is to develop reasoning mechanisms that deliberate over raw values rather than objectives. Similarly, Yokoo and colleagues formalize negotiation as an extension to the classic single agent constraint satisfaction framework [60, 62], where variables and constraints are distributed among multiple agents. Search algorithms (asynchronous backtracking and asynchronous weak-commitment search) are shown to solve this distributed problem. Both algorithms are complete and the asynchronous weak-commitment is shown to be more efficient. However, although concerned with the computational tractability of negotiation, the agents' search problem is simplified through resolution over only a single variable *and* the implicit assumption that agents communicate constraints and modify their local solutions cooperatively. Even when multiple variables are considered [61], the second assumption greatly helps the search process. However, in open systems, agents are motivated to misrepresent their true constraints for selfish reasons. Our trade-off algorithm implements a distributed multi-issue constraint modification strategy that requires no such explicit communication of constraints. Furthermore, since similarity heuristics can lead to deals with higher social welfare then rational agents are better off using such a decision mechanism. In this model the similarity heuristic captures the strategic element of decision making; more successful outcomes can be expected for those decisions that increase the similarity of two demands. Therefore, agents are better off in the horizon of the game when they invest time and computation in maximising the similarity metric.

Finally, although similarity is a basic tool in at least three cognitive tasks (classification, case-based reasoning, and interpolation) it has received comparatively little attention in the context of logical models of reasoning. It has, however, been used in work on psychological studies of human behaviour [55], mathematical work on graded extensions of equivalence relations [56, 64], and as a model of approximate reasoning [22, 59]. From the philosophical perspective, Niiniluoto relates similarity with the broader area of analogical reasoning [34]. Finally, although similarity has been frequently used to model case-based reasoning, it has never been used to model negotiation processes between autonomous agents.

6 Conclusions and Future Work

This paper presented a formal heuristic model and a particular linear algorithm for performing trade-offs in automated negotiations. Based on our experiences with a number of real-world applications, the algorithm had to be designed to work in a distributed setting in which the agents have limited information about the preferences of their negotiation opponent, limited computational resources to devote to the negotiation process, and limited opportunities for repeated encounters. For these reasons, we decided the notion of similarity should be the cornerstone of our trade-off approach since this enables the agents to model the domain of the negotiation decision variables rather than the specifics of their negotiation opponent. The particular technique we adopted was fuzzy similarity since this enables us to cope with the inherent uncertainties in the negotiation process. From this basis, we developed a novel hill-climbing algorithm for performing trade-offs in multi-dimensional negotiations that involve both qualitative and quantitative decision variables. We analysed the algorithm theoretically and found its average complexity to be linearly proportional to the number of negotiation decision variables under consideration. Moreover, our empirical evaluation demonstrated the algorithm's effectiveness in generating trade-offs in a range of negotiation contexts. Specifically we showed that as our algorithm explores more of the set of possible outcomes so it produces agreements that have higher joint gains. This increased search results in: (i) higher joint outcomes on each iteration of the algorithm, across a single run in a unique environment or across multiple environments; and (ii) higher communication costs since more proposals are exchanged before an agreement is reached.

For the future, there are four broad directions in which this research can be extended. Firstly, we would like to develop a more sophisticated meta-strategy controller. In particular, we would like to develop an intelligent controller that can select the negotiation strategy according to the agent's prevailing context and its negotiation objectives. Such a meta-controller would be able to decide when it is appropriate to engage in a trade-off negotiation, when it is appropriate to disengage from a trade-off negotiation, which of the negotiation decision variables should be subject to trade-offs at the current time, and how to set the various parameters of the trade-off algorithm in order to optimise the agent's performance. Secondly, we would like to explore the opportunities for an agent to learn information about its negotiation opponent so that the agents can come to higher quality agreements in a more efficient manner. In particular, learning information about the opponent's preferences and their relative weightings is likely to lead to better outcomes. The third future goal of our research is to evaluate the current algorithm and the above proposed extensions against other negotiation algorithms. Finally, we aim to design and evaluate other algorithms for computing trade-offs when agent's utility models are assumed to be non-linear. Pareto-optimality of distributed global optimisation algorithms, such as tabu search and simulated annealing, are currently being evaluated in the context of a distributed optimisation/negotiation for complex non-linear games [10].

7 Acknowledgements

Some of the material of this paper is the result of collaborative work with different researchers among which we'd like to mention Lluís Godo. This line of research is currently being supported by the MCYT research project eINSTITUTOR (TIC2000-1414).

References

[1] M. Barbuceanu and W. Lo. A multi-attribute utility theoretic negotiation architecture for electronic commerce. In *Proceedings of the Fourth International Conference on Autonomous Agents, Barcelona, Spain (Agents-2000)*, pages 239–247, Barcelona, Spain, 2000.

[2] K. Binmore. *Essays on the foundations of game theory*. Basil Blackwell., Oxford, UK, 1990.

[3] K. Binmore. *Fun and Games: A Text on Game Theory*. D.C. Heath and Company., Lexington, Massachusetts, 1992.

[4] C. Castlefranchi and R. Conte. Limits of strategic rationality for agents and MA Systems. In M. Boman and W. Van de Velde, editors, *Multi-Agent Rationality: Proceedings of the 8th European Workshop on Modeling Autonomous Agents in Multi-Agent World, MAAMAW'97*, number 1237 in Lecture Notes in Artificial Intelligence. Springer-Verlag, 1997.

[5] S.H. Clearwater. *Market Based-Control*. World Scientific, 1996.

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

[7] H. Ehtamo, E. Ketteunen, and R. Hamalainen. Searching for joint gains in multi-party negotiations. *European Journal of Operational Research*, 1(30):54–69, 2001.

[8] P. Faratin. *Auotmated Service Negotiation between Autonomous Computational Agents*. PhD thesis, Department of Electronic Engineering, Queen Mary and Westfield College, University of London, 2000.

[9] P. Faratin, N. R. Jennings, P. Buckle, and C. Sierra. Automated negotiation for provisioning virtual private networks using fipa-compliant agents. In *Proceedings of the Fifth International Conference on The Practical Application of Intelligent Agents and Multi-Agent Technology (PAAM-2000), Manchester, UK*, pages 185–202, 2000.

[10] P. Faratin, M. Klein, H. Samaya, and Yaneer Bar-Yam. Simple negotiating agents in complex games: Emergent equilibria and dominance of strategies. In John-Jules Meyer and M. Tambe, editors, *Proceedings of the 8th Int Workshop on Agent Theories, Architectures and Languages (ATAL-01), Seattle, USA*, LNCS, pages 42–53. Springer Verlag, 2001.

[11] P. Faratin, C. Sierra, and N. R. Jennings. Using similarity criteria to make negotiation trade-offs. In *Proceedings of the 4th Int. Conf. on Multi-Agent Systems*, pages 119–126, Boston, USA., 2000.

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

[13] R. Gibbons. *A Primer in Game Theory*. Harvester Wheatsheaf, New York, 1992.

[14] P. J. Gmytrasiewicz and E.H. Durfee. A logic of knowledge and belief for recursive modeling: Preliminary report. In *In Proceedings of the Tenth National Conference on Artificial Intelligence*, pages 628–634, 1992.

[15] Piotr J. Gmytrasiewicz and E.H. Durfee. Elements of a utilitarian theory of knowledge and action. In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence, Chambrey, France*, pages 396–402, 1993.

[16] S. Gupta and Z. Livine. Resolving a conflict situation with a reference outcome: An axiomatic model. *Management Science*, 34(11):1303–1314, 1988.

[17] P. Heiskanen. Decentralized method for computing pareto solutions in multiparty negotiations. *European Journal of Operational Research*, 117:578–590, 1999.

[18] N. R. Jennings, P. Faratin, A.R. Lomuscio, S. Parsons, C. Sierra, and M. Wooldridge. Automated negotiation: prospects, methods and challenges. *International Journal of Group Decision and Negotiation*, 10(2):199–215, 2001.

[19] N. R. Jennings, P. Faratin, T. J. Norman, P. O'Brien, and B. Odgers. Autonomous agents for business process management. *Int. Journal of Applied Artificial Intelligence.*, 14(2):145–189, 2000.

[20] N.R. Jennings. An agent based approach for building complex systems. *Communication of ACM*, 44(4):35–41, 2001.

[21] E. Kalai and M. Smorodinsky. Other solutions to nash's bargaining problem. *Econometrica*, 43:513–518, 1975.

[22] F. Klawon and V Novak. The relation between inference and interpolation in the framework of fuzzy systems. *Fuzzy Sets and Systems*, 81(3):331–354, 1996.

[23] S. Kraus. Negotiation and cooperation in multi-agent environments. *Artificial Intelligence*, 94(1–2):79–97, 1997.

[24] S. Kraus and D. Lehmann. Designing and building negotiation automated agent. *Computational Intelligence*, 11(1):132–171, 1995.

[25] K. Larson and T.W. Sandholm. Barganing with limited computation: Deliberation equilibrium. *Artificial Intelligence*, 132(2):183–217, 2001.

[26] D.G. Luenberger. *Introduction to linear and nonlinear programming*. Addison-Wesley, 1973.

[27] K.R. MaCrimmon and D.M. Messick. A framework for social motives. *Behavioural Science*, 21:86–100, 1976.

[28] D. Marr. *Vision : a computational investigation into the human representation and processing of visual information*. W.H. Freeman, San Francisco, 1982.

[29] R.B. Myerson. *Game Theory: Analysis of Conflict*. Harvard University Press, Cambridge, MA, 1991.

[30] J. F. Nash. The bargaining problem. *Econometrica*, 18:155–162, 1950.

[31] J. F. Nash. Non-cooperative games. *Annals of Mathematics*, 54(2):286–295, 1951.

[32] J.F. Nash. *Non-cooperative Games*. PhD thesis, Mathematics Department, Princeton University, 1950.

[33] I. Newton. Principia mathematica. In A. Koyr and I.B Cohen, editors, *Isaac Newton's Philosophiae Naturalis Principia Mathematica*, Cambridge, Mass., 1972. Harvard University Press,.

[34] I. Niiniluoto. *Analogy and Similarity in Scientific Reasoning*, volume 197 of *Studies in Epistemology, Logic, Methodology, and Philosophy of Science*, pages 271–298. Kluwer, 1988.

[35] P. Noriega and C. Sierra. Auctions and Multi-agent Systems. In M. Klusch, editor, *Intelligent Information Agents*, pages 153–175. Springer, 1999.

[36] D. Parkes. Iterative combinatorial auctions: Theory and practice. In *Proc. Seventeenth National Conference on Artificial Intelligence, Austin, Texas, July 30-August 3*, 2000.

[37] S. Parsons, C. Sierra, and N. R. Jennings. Agents that reason and negotiate by arguing. *Journal of Logic and Computation*, 8(3):261–292, 1998.

[38] W. Pedrycz and F. Comide. *An Introduction to Fuzzy Sets: Analysis and Design*. MIT Press, Cambridge, Massachusetts, 1998.

[39] D.M Pennock and M.P Wellman. Representing aggregate belief through the competitive equilibrium of a securities market. In *Proceeding of the Thirteenth Conference on Uncertainty in Artificial Intelligence*, pages 392–400, 1997.

[40] D.G. Pruitt. *Negotiation Behavior*. Academic Press, New York, 1981.

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

[42] J.A. Rodríguez, P. Noriega, C. Sierra, and J.A. Padget. FM96.5 a java-based electronic auction house. In *In Second International Conference on The Practical Application of Intelligent Agents and Multi-Agent Technology: (PAAM-97), London, UK*, pages 207–224, 1997.

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

[44] A. E. Roth. Toward a focal point theory of bargaining. In A. E. Roth, editor, *Game Theoretic Models of Bargaining*, pages 259–268. Cambridge University Press, NY, 1985.

[45] A. Rubinstein. Perfect equilibrium in a bargaining model. *Econometrica*, 50:97–109, 1982.

[46] A. Rubinstein. Toward a focal point theory of bargaining. In A.E. Roth, editor, *Game-Theoretic Models of Bargaining*, pages 99–114. Cambridge University Press, NY, 1985.

[47] S. Russell and P. Norvig. *Artificial Intelligence: A modern approach*. Prentice Hall, Upper Saddle River, New Jersey, 1995.

[48] T.W. Sandholm and V.R. Lesser. Coalitions among computationally bounded agents. *Artificial Intelligence*, 94(1):99–137, 1997.

[49] A. Sathi and M.S. Fox. Constraint-directed negotiation of resource reallocation. In L. Gasser and M. Huhns, editors, *Distributed Artificial Intelligence Volume II*, pages 163–195, San Mateo, California, 1989. Morgan Kaufmann.

[50] 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, *Proceedings of 8th European Workshop on Modeling Autonomous Agents in Multi-Agent World*, number 1237 in Lecture Notes in Artificial Intelligence, pages 17–35. Springer-Verlag, 1997.

[51] H. A. Simon. *The Sciences of the Artificial*. MIT Press, Cambridge, Massachusetts., 1996.

[52] R.G. Smith. The contract net protocol: High-level communication and control in a distributed problem solver. *IEEE Transactions on Computers*, C-29(12):1104–1113, 1980.

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

[54] J.E. Teich, H. Wallenius, J. Wallenius, and S. Zionts. Identifying pareto-optimal settlements for two-party resource allocation negotiations. *European Journal of Operational Research*, 93:536–549, 1996.

[55] A. Tversky. Features of similarity. *Psychological Review*, 84(4):327–352, 1977.

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

[57] N. Vulkan and N. R. Jennings. Efficient mechanisms for the supply of services in multi-agent environments. *Int Journal of Decision Support Systems*, 28(1–2):5–19, 2000.

[58] M. P. Wellman and P. R. Wurman. Market-aware agents for a multiagent world. *Robotics and Autonomous Systems*, 24(3–4):115–127, 1998.

[59] M. Ying. A logic for approximate reasoning. *Journal of Symbolic Logic*, 59(3):830–837, 1994.

[60] M. Yokoo, E.H. Durfee, T. Ishida, and K. Kuwabara. The distributed constraint satisfaction problem: Formalization and algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 10(5):673–685, 1998.

[61] M. Yokoo and K. Hirayama. Distributed constraint satisfaction algorithm for complex local problems. In *Proceedings of the Third International Conference on Multi-Agent Systems*, pages 372–379. IEEE Computer Society Press, 1998.

[62] M. Yokoo and T. Ishida. Search algorithms for agents. In G. Weiss, editor, *Multiagent Systems*, pages 165–201. The MIT Press, Cambridge, Massachusetts, 1999.

[63] O. R. Young. *Bargaining: Formal Theories of Negotiation*. University of Illinois Press, Urbana, 1975.

[64] L. A. Zadeh. Similarity relations and fuzzy orderings. *Information Sciences*, 3:177–200, 1971.