Auton Agent Multi-Agent Syst (2007) 15:221–252
DOI 10.1007/s10458-006-9009-y

An agent architecture for multi-attribute negotiation using incomplete preference information

Catholijn M. Jonker · Valentin Robu · Jan Treur

Published online: 10 January 2007

Abstract A component-based generic agent architecture for multi-attribute (integrative) negotiation is introduced and its application is described in a prototype system for negotiation about cars, developed in cooperation with, among others, Dutch Telecom KPN. The approach can be characterized as cooperative one-to-one multi-criteria negotiation in which the privacy of both parties is protected as much as desired. We model a mechanism in which agents are able to use any amount of incomplete preference information revealed by the negotiation partner in order to improve the efficiency of the reached agreements. Moreover, we show that the outcome of such a negotiation can be further improved by incorporating a "guessing" heuristic, by which an agent uses the history of the opponent's bids to predict his preferences. Experimental evaluation shows that the combination of these two strategies leads to agreement points close to or on the Pareto-efficient frontier. The main original contribution of this paper is that it shows that it is possible for parties in a cooperative negotiation to reveal only a limited amount of preference information to each other, but still obtain significant joint gains in the outcome.

C. M. Jonker
Faculty of Electrical Engineering, Mathematics and Computer Science,
Delft University of Technology,
Mekelweg 4, 2628 CD Delft, The Netherlands
e-mail: catholijn@mmi.tudelft.nl

V. Robu (✉)
CWI, Dutch Research Center for Mathematics and Computer Science,
Kruislaan 403, 1098 SJ Amsterdam, The Netherlands
e-mail: robu@cwi.nl

J. Treur
Department of Artificial Intelligence,
Vrije Universiteit Amsterdam,
De Boelelaan 1081a, 1081 HV Amsterdam, The Netherlands
e-mail: treur@cs.vu.nl

Keywords Agent architecture · Bilateral negotiation · Multi-attribute negotiation · Cooperative negotiation · Component-based modeling · Cognitive modeling

1 Introduction

Recent years have shown a surge of interest in negotiation technologies, seen as a key coordination mechanism for the interaction of providers and consumers in future electronic markets that transcend the selling of uniform goods [20,31]. Suggested applications range from modeling interactions between customers and merchants in retail electronic commerce [13], to the online sale of information goods [24], or reducing operational procurement costs of large companies [1]. Such technologies could prove especially useful in the case of multi-attribute negotiations, where the agents have an incentive to cooperate in order to search for an outcome that brings joint gains for both parties. As shown by Ref. [28], such negotiations represent non-zero sum games, where "as values shift along multiple directions it is possible for both parties to be better off". In these settings, agents often care about equity and social welfare, and not only about their own individual utility [10]. Examples where such cases may arise are: business process management involving agents within the same organization [10] or e-commerce negotiations where the seller is interested in having a satisfied buyer [13].

Gutman and Maes [13] discuss the difference between competitive and cooperative negotiation models. They show that the competitive negotiations in retail markets are unnecessarily hostile to customers and offer no long-term benefits to merchants. Essentially, in competitive negotiations the merchant is pitted against the customer in price-tug-of-wars. They conclude that merchants often care less about profit on any given transaction and care more about long-term profitability, which implies customer satisfaction and long-term customer relationships. Their analysis makes a strong case for cooperative negotiation for the retail market.

Regarding the problem from a game-theoretic perspective, the main problem that arises is that cooperative game theory generally assumes that complete information of both parties is available in order to compute optimal outcomes. This does not hold for many applications, where only a limited degree of trust exists between parties in sharing preference information. The reasons for this may be endogenous to the negotiation (e.g. fear that the other may abuse this information to get a better deal) or exogenous (e.g. privacy concerns).

In classical multi-attribute-utility theory (see e.g. [19,25,26]), the solution proposed is the use of an independent mediator, which both parties can trust to reveal their preferences. The problem with this approach in an electronic or open system setting is that it can be difficult to establish whether a mediator is indeed impartial or more trustworthy than the negotiation partner himself. For example, an agent may have no way of knowing if the solutions proposed by the mediator are not biased towards the other or that his preference information will not be stored and used for other purposes. By contrast, our approach is to use a distributed design, in which each agent computes its own bids, using the information available about the preferences of the opponent. We take into account two different types of (incomplete) information:

- Partial profile information, which is communicated by the negotiation partner herself in the beginning of the negotiation.

- Profile information which can be deduced (learned) from successive bids during the negotiation itself. Here we start from the assumption that the way the negotiation partner is bidding may reveal something about his preferences. For this mechanism we use the term "guessing" to clearly show it is a heuristic.

In our work we preferred the heuristic approach to designing automated negotiation, since we feel this allows more flexibility. This position is supported, among others, by Ref. [10] who clearly show that "what is required are agent architectures that implement different search mechanisms, capable of exploring the set of possible outcomes under both limited information and computation assumptions". However, this does not mean we ignore the results from game theory: they are present in both measuring the efficiency of reached agreements (e.g. Pareto-efficiency) and in analyzing some properties of our mechanism.

The rest of this paper is organized as follows. Section 2 discusses multi-attribute negotiation in the more general framework of processes in which such a negotiation occurs. The section also describes a general system that incorporates the negotiation model as introduced in this paper. The general system is a brokering system in which consumers and providers are matched together given the demands from the consumers and possible offers from the providers. Section 3 presents the formal design of the negotiation model. Experimental validation of the negotiation model is the topic of Sect. 4. Section 5 discusses the results of our model and compares it other approaches in literature. The conclusion is presented in Sect. 6.

2 Brokering and negotiation

In this section, the topic of negotiation is placed within the greater context of the consumer buying behaviour model (CBB) of Gutman and Maes [13]. The second part of this section describes a generic brokering model that covers the first stages of the CBB model, with negotiation being the last stage covered.

2.1 Modeling consumer behavior with agents

The process of brokering as often occurs in electronic commerce involves a number of agents. A user offering products may be supported by a provider agent that provides information about the products to other (human or computer) agents. A user looking for products may be supported by a personal assistant agent that takes its user's queries and contacts other agents or looks at the Web directly to find information on products within the user's scope of interest. Such a personal assistant agent may contact either provider agents immediately, or mediating agents, which in turn have contact with provider agents, or other mediating agents. Depending on the application, the chain of agents involved may include zero or more mediating agents.

The Consumer buying behaviour model (CBB) (see [13]) consists of six main stages: Need Identification, Product Brokering, Merchant Brokering, Negotiation, Purchase and Delivery, and Service and Evaluation. The model discussed in this paper addresses the first four of these stages, where the product brokering is an integrated part of the entire brokering process and overlaps with the need identification. This is in line with normal procedures, as "CBB stages often overlap and migration from one to another is sometimes non-linear and iterative". The buyer contacts the broker agent, the broker agent provides the buyer with forms in order to determine the wishes of the buyer. Then the broker matches products and suppliers against the wishes of the buyer presenting him with the best three options. The buyer can then select one of these proposals. A special buyer representative agent negotiates with the representative agent of the supplier to obtain the best configuration of the selected option. The different attributes of the object under negotiation, the possible values for each of those attributes, and the different wishes (profiles) of consumer and provider, allow for cooperative negotiation: cooperative negotiation can be seen as a decision-making process of resolving a conflict involving two or more parties over multiple interdependent, but non-mutually exclusive goals, cf. [22].

2.2 A Brokering System

In Ref. [16], a design for a generic multi-agent model for brokering is introduced based on a Generic Agent Model GAM that was developed using the component-based agent design method DESIRE (see [4]).

The multi-agent system in which the negotiation models can and have been applied consists of the following types of agents, see Fig. 1: Human Buyers, Human Dealers, Buyer Representative agents, Dealer Representative agents, Broker agent. Moreover, to model retrieval of information from databases, a number of components is used; one of them is the External World from which Buyer Representative agents can retrieve third party information (the actual third party chosen depends on the type of items being brokered like consumer organizations and the Land registry office). Furthermore, specific Seller-dependent databases are included, from which the Seller Representative agent can retrieve additional information about the item offered.

The generic agent architecture for multi-attribute negotiation and for the broker agent was designed and formally specified using DESIRE, as a refinement of the Generic Agent Model, GAM, see Ref. [4]. The processes modeled within the generic agent model are depicted in Figs. 2 and 3. The processes involved in controlling an agent (e.g. determining, monitoring and evaluating its own goals and plans) but also the processes of maintaining a self-model are the task of the component Own Process Control. The processes involved in managing communication with other agents are the task of the component Agent Interaction Management. Maintaining knowledge of other agents' abilities and knowledge is the task of the component Maintenance

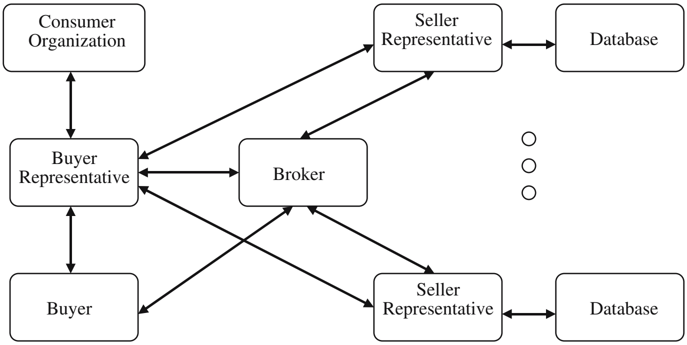


Fig. 1 The Brokering System

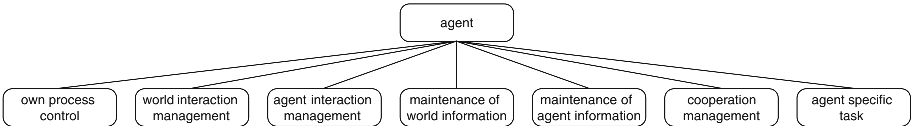

agent
own process control
world interaction management
agent interaction management
maintenance of world information
maintenance of agent information
cooperation management
agent specific task

Fig. 2 Processes at the two highest process abstraction levels within the agent

of Agent Information. Comparably, the processes involved in managing interaction with the external (material) world are the task of the component World Interaction Management. Maintaining knowledge of the external (material) world is the task of the component Maintenance of World Information. The specific task for which an agent is designed (for example: design, diagnosis, information retrieval), is modeled in the component Agent Specific Task. Existing (generic) task models may be used to further structure this component. In addition, a component Cooperation Management may be distinguished for all tasks related to social processes such as cooperation in a project, or negotiation. This component is discussed in the next section.

The rest of this section describes, in greater details, the internal design of the broker agent, for the highest internal level of composition see Fig. 3. Here specific broker capabilities are modelled within the components cooperation Management and Agent Specific Tasks.

At the highest abstraction level within an agent, a number of processes can be distinguished that support interaction with the other agents. First, a process that manages communication with other agents, modeled by the component Agent Interaction management in Fig. 3. This component analyses incoming information and determines which other processes within the agent need the communicated information. Moreover, outgoing communication is prepared. Next, the agent needs to maintain information on the other agents with which it cooperates: Maintenance of Agent Information. The component Maintenance of World Information is included to store the information on world information (e.g. information on attributes of products). The component Own Process Control defines different characteristics of the agent and determines foci for behavior. The component World Interaction Management is included to model interaction with the world. The negotiation capabilities of the agent (which are the main focus of this paper) are, in the current model, placed in the component Cooperation Management.

In the remainder of this paper, we will focus our attention on a part of this large brokering framework, namely on modeling the bilateral negotiation between the Buyer Representative and Seller Representative agents. Other aspects of this framework have been considered, but they have been presented and discussed in further detail elsewhere (e.g. [4,16]).

3 The negotiation model

The negotiation considered follows an alternating-offers protocol; for more information on protocols for negotiation, see Ref. [20]. A bid in such a negotiation has the form of values assigned to a number of attributes. If the negotiation is about the sale of a car, the relevant attributes considered are, for example: CD player, extra speakers, airco, tow hedge, price and then a bid consists of an indication of which CD

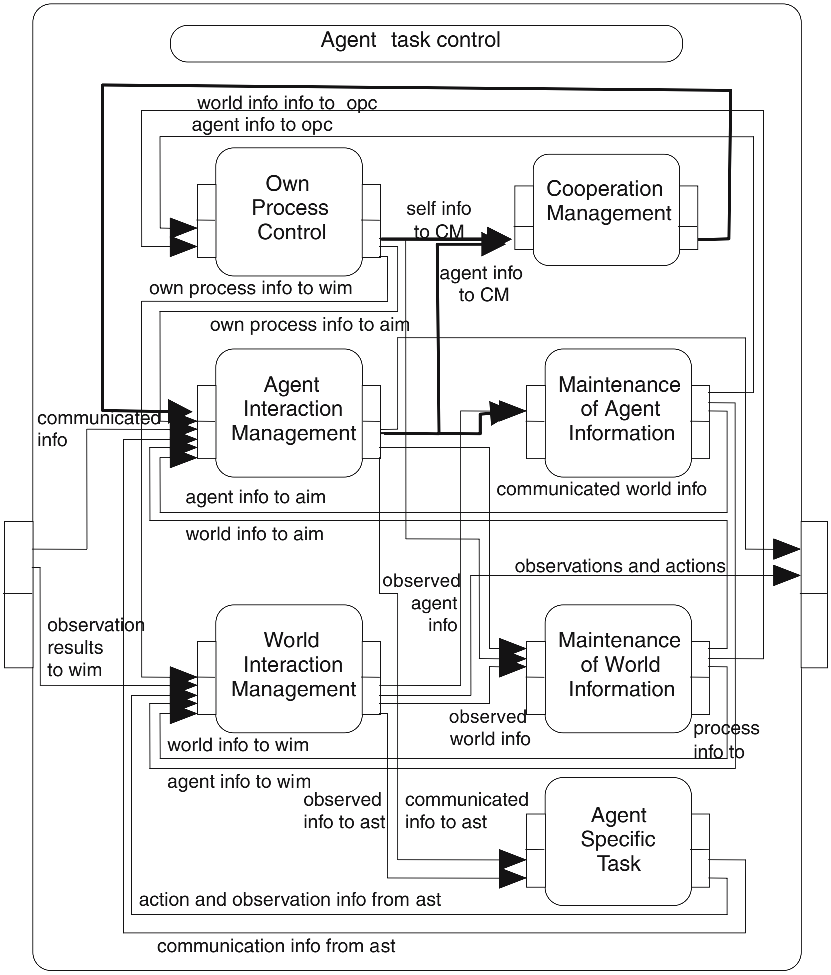


Fig. 3 Top level composition of the broker agent

player is meant, which extra speakers, airco and tow hedge, and what the price of the offer is. Although the examples and discussion provided in this paper are based on this domain, our negotiation model is a generic one and this section provides a generic formal description of the model. Instantiations in other domains are possible and have been considered.[1]

To assess a bid of the other party, it is important to have evaluation methods. Evaluation can be done at two levels: the level of each of the specific attributes (attribute evaluation), and the level of the bid as a whole (overall bid utility). Taking this into account, some characteristics of the multi-attribute negotiation model presented here are:

- explicit reasoning about the negotiation strategy and coordination of the negotiation process

[1] For example, modeling negotiations between employer and employee regarding work shifts and overtime pay (work performed in collaboration with Almende B.V., Rotterdam).

- evaluation of a bid takes into account both the attributes separately and the overall utility of the bid
- planning of a new bid takes into account both the overall utility level and the level of attributes separately

In particular, in the model it is possible to work on two levels: the level of the overall bid, and the level of each of the attributes separately. The negotiation model has been specified as a compositional structure within the component Cooperation Management of the GAM, see Sect. 2. Globally speaking, the process runs as follows:

- For each negotiation round, first evaluations of the attributes of the previous bids are determined.
- Then these evaluations are aggregated and the overall utilities are computed, both for the own previous bid of the agent and the proposed bid, received from the opponent in the previous negotiation step.
- Next, the concession step to be made in the next bid is determined, w.r.t the utilites of the previous bids. Thus, the agent determines the overall target utility for the next bid to be proposed to the opponent.
- To obtain the next bid, given the target utility, first according to some distribution over attributes, target attribute evaluation values are determined (chosen in such a manner that they aggregate exactly to the target utility)
- Finally, for each of these target attribute evaluation values, an attribute value is chosen that has an evaluation value as close as possible to the target evaluation value for the attribute.

In the last step, if only discrete attribute values exist, it may be the case that the target utility is not reached. However, if at least one of the attributes has continuous values, then this attribute can be chosen to compensate for differences that are created due to the mapping to discrete values for the other attributes. In our application, the price attribute is such a continuous attribute, and chosen to compensate for differences. In this manner bids are created that exactly match the target utilities. To realize the compositional process structure sketched above, at its top level the component Cooperation Management is composed of the five components in Fig. 4: Negotiation Coordination, Attribute Evaluation, Bid Utility Determination, Utility Planning, and Attribute Planning. In the rest of Sect. 3, the presentation will follow this design structure and that of the Attribute Planning sub-component, which can be further composed as shown in Figs. 5 and 6.

3.1 Negotiation coordination

Within the component Negotiation Coordination the negotiation process state is analyzed (component Process Analysis) and the process is controlled (component Process Control). Process Analysis determines which of the following are true and which are false:

(a) *Repetition of steps* takes place: steps without enough progress (depending on the impatience factor (π) which specifies the acceptable number of steps in which nothing changes)
(b) A *utility gap* (larger than some threshold ω) remains; i.e. a significant difference between the utility of the own bid and that of the other agent's bid.

Fig. 4 Multi-Attribute Negotiation Architecture (corresponding to the internal composition of the Cooperation Management component in Fig. 3)

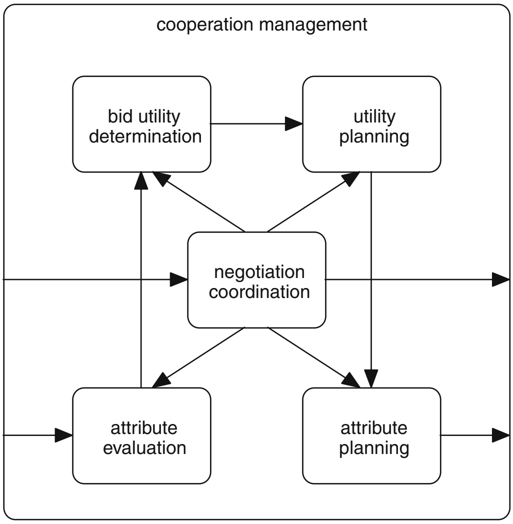
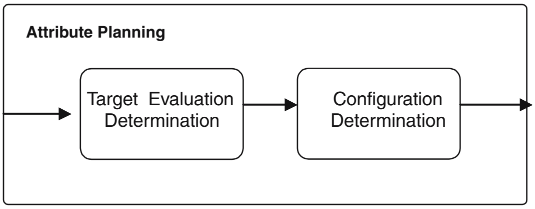


Fig. 5 Internal composition of Attribute Planning, for the case no explicit opponent modeling is used



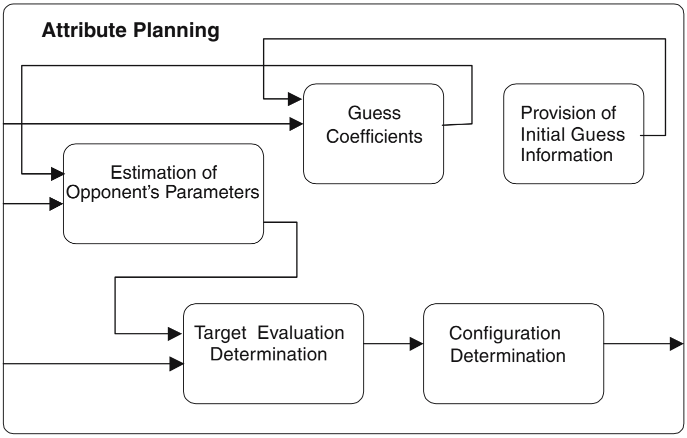

Attribute Planning
Guess Coefficients
Provision of Initial Guess Information
Estimation of Opponent's Parameters
Target Evaluation Determination
Configuration Determination

Fig. 6 Internal composition of Attribute Planning sub-component, in case modeling of opponent preferences is used

(c) A *configuration mismatch* (larger than some threshold ν) remains between the own bid and the other agent's bid.

Here a configuration mismatch means that for at least one attribute, between the two values (in the two bids) a significant difference exists. Depending on the outcome of the analysis within component Process Control the following actions can be decided upon:

1. Start a *next negotiation round*
2. Contact the user to discuss whether the *concession factor* (γ) can be changed.
3. Contact the user to discuss whether the *configuration tolerance* (τ) can be changed.
4. Communicate to the user that an *agreement* has been reached.
5. Communicate to the user that the *negotiation has failed* (only when the user is unwilling to change the characteristics).

3.2 Attribute evaluation

The evaluation functions for each attribute take either a *table form* or another *specific function description*. A table form is used for discrete attributes such as accessories. Specific function descriptions are used for continuous attributes such as mileage or price. The form of specific function descriptions are of a type such as 'linear', or 'uphill'. For the attributes for which a specific (non-table) type of evaluation function is given, depending on this type, knowledge is specified to obtain the object evaluations. Currently, only specific function types are used that consist of linear parts, cut-off between 0 and 1: linear function, normal distribution function, downhill function, uphill function.

If desired, in all evaluations and utilities, the model supports that two aspects can be modeled separately and integrated: *ease evaluation* and *ease utility* EU and *financial evaluation* and *financial utility* FU. The latter aspect covers the financial rationality in the agent's behavior. The former aspect models all other aspects within the decision making such as a resistance against more complicated transactions (even if in terms of economic gain they are more favorable).

From a utility function perspective, the above definition means that utility functions of the agents in our model are not assumed to be quasi-linear, i.e. there is not implicit assumption that a utility value can be mapped directly into an amount of money (as a number of other negotiation and auction models do). In our model, the utility of a multi-attribute contract is always mapped to a value between 0 and 1. Thus, both the configuration of the attributes (in our case the quality of the car accessories offered) and the price (amount to be paid or received for the car) are evaluated, by mapping them to a value between 0 and 1. The balance between these two aspects within the overall evaluations is defined by the *financial rationality factor* ρ. If this factor is 1, then only the financial utility is taken into account (i.e. the agent is completely financially driven), if it is 0, only the ease utility (completely ease driven). Any factor in between 0 and 1 defines the relative weight of the economic aspect compared to the ease aspect in the decision making.

We acknowledge that mapping a sum of money through a utility function is not a standard choice in all negotiation models (though it is not unusual, since we follow the seminal work on multi-attribute negotiation of Raiffa [25]). This allows us more modeling flexibility. In the car dealership example, some buyers (with a better financial position) do not assign such a high utility to an additional amount of money for accessories, while others assign a much larger utility to the money than to ease/convenience. Similarly, some the sellers (i.e. car dealers) assign a higher utility to the difficulty of procuring installing the accessories required by the buyer (the ease factor), while for others this is not such an important consideration. In the following, we formalize these intuitions and give more formal definition of utility functions.

3.3 Bid utility determination

This section introduces the utility model used and formally defines the utility functions. The computing (of the utility value of each bid) is performed in the Bid Utility Determination sub-component of the model (c.f. Fig. 4).

The model proposed here is symmetrical for both parties, i.e. both buyer and seller use the same type (i.e. mathematical form) of linear utility functions, as defined below. However, the actual utility function and parameters are different for buyer and seller (and are specified by the human owners of the agents through a software interface, before the negotiation begins)[2] and are *private information* for each party during the negotiation. Thus, unless otherwise described, an agent can only be assumed to know its own utility parameters, which in the general case, may be very different from the parameters of the negotiation partner (these remain private information).

From the point of view of notation, all parameters defined in this section could be further indexed with the subscripts "buyer" and "seller" (or, from the perspective of each agent, they could be indexed with "own" and "other"). However, in computing its own utility value for a bid, only the own utility parameters of the agent are needed (the opponent's utility parameters are not known and actually not needed, since the agent only uses its own utility function). Therefore, in this section we do not use an index identifying the agent for all the parameters. However, in further sections, when computing how to make attribute-specific concessions, the agent will need to approximate the opponent's preference weights, and thus we will introduce an additional index to distinguish between the self/opponent parameters.

As mentioned, the utility of one bid is divided into computing the ease utility and the financial utility, and we discuss these in different sections.

3.3.1 Ease utility determination

The ease utility is computed for all the attributes, except the price, and it reflects a subjective preference of each of the agents. Within the model, the *ease utility* EU_B of a bid B is taken as a weighted sum of the *attribute evaluation values* $E_{B,j}$ for the different negotiant attributes denoted by j.

$$EU_B = \Sigma_j w_j E_{B,j} \tag{1}$$

The parameters $E_{B,j}$ are specified by the human user of the agent, through an interface, with values between 0 and 100 (which are taken as values between 0 and 1, in the model, by division by 100).

The weights w_j are relative importance factors based on the *importance factors* p_k for the different attributes:

$$w_j = p_j / \Sigma_k p_k \tag{2}$$

The importance factors for each attribute p_j (between 0 and 100) are also specified by the user (owner) of the agent through an interface. Formula (2) insures that the sum of the attribute importance weights for one agent is always normalized to be equal to one, regardless of the inputs of the human user, i.e. $\Sigma_k w_k = 1$.

[2] A site with screen shoots of the software tool used to input these values (initially prepared for a software demonstration at the AAMAS'04 conference), is available online at: http://homepages.cwi.nl/~robu/aamas/aamas_demo.html

The rationale for defining the preference weights this way (indirectly, through importance factors) is that we found that it is more intuitive for human users of the system to specify a number between 0 and 100 through an interface, without worrying about having the values add up to 1 (or respectively 100), and allow the agent to do the normalizations automatically.

To illustrate, in the car dealership domain, there are 4 discrete attributes taken into consideration: Airco, CD player, Tow Hedge and Extra Speakers. Each of these can take 5 quality levels (denoted by $E_{B,j}$ in the Formula 1) as: none, meagre, standard, good, very good. Each of these is assigned, for each attribute by both Buyer and Seller, through their private interface, a value from 0 to 100 (taken as between 0 and 1). Each attribute is also assigned an importance factor, from which a weight is computed such that the sum of weight normalizes to add up to 1. Thus, from these values the ease utility between 0 and 1 can be computed for any bid or contract combination, from the perspective of each agent.

3.3.2 Financial utility determination

If a financial utility is used separately, then the above utility (called the *ease utility* EU_B) is determined on the basis of all attributes except price. *Financial utility* FU_B is based on the financial balance g_B for a given bid B:

$$g_B = p_B - b - a_B \tag{3}$$

where b denotes the *basic costs* (the cost of the object without additional accessories) and a_B denotes the *additional costs* of bid B, and p_B price within bid B.

$$a_B = \Sigma_j FE_{B,j} \tag{4}$$

that is based on the financial evaluations ($FE_{B,j}$) of the values of the different attributes j. However, to be able to relate FU_B to the ease utility EU_B, FU_B is the normalization of the financial balance to a number between 0 and 1:

$$FU_{B,j} = g_B/\delta b \tag{5}$$

The fraction δ is the fraction of the basic cost that is maximally additionally (to be) earned (e.g. .3, a *maximum margin* of 30%). Some notes can be made.

- The financial utility FU is defined on the interval between 0 and 1 in such a manner that financial utility 1 means cost price plus maximal margin ($b + \delta b + a_B$).
- Let B_0 be the initial bid of the seller, then by taking price $p_{B_0} = b + \delta b + a_{B_0}$, the financial utility of this bid is

$$FU_{B_0} = (p_{B_0} - b - a_{B_0})/\delta b = 1. \tag{6}$$

- By setting δ properly, the seller makes sure that (s)he is not asking unrealistic prices.
- The financial utility is defined on the interval between 0 and 1 in such a manner that $FU_B = 0$ implies $p_B = b + a_B$, i.e. the cost price. So, if a buyer makes a bid B with $p_B < b + a_B$, then $FU_B < 0$ from the perspective of the seller.

Thus, we specify the financial utility function as a linear, continuous, function between two cut-off points: a maximum and a minimum financial margin (also called "mark-up"), over the costs (see also end of Sect. 3.4). This choice fits the business model of many sellers operating in retail commerce: they receive the goods at a certain cost and have a minimum profit margin (below which it is unprofitable to run the business) and a maximum profit margin they expect to make on the sale of good. (the particular example considered in this paper is about the sale of a car, but this holds for many other retail domains). In our model, selling at the cost price corresponds to receiving zero financial utility, while selling it with a maximum expected profit corresponds to a financial utility of 1.

The seller agent is designed to start the negotiation by asking a price corresponding to the maximal expected margin. However, it will not concede on a price that falls below a minimum acceptable financial utility margin, denoted by ε (see Sect. 3.4 for a formal presentation of the concession model).

On the basis of the ease utility and the financial utility, the *overall utility* is determined as a weighted sum. Here the weights are based on the financial rationality factor ρ (part of the dealer profile).

$$U_B = \rho FU_B + (1 - \rho)EU_B \tag{7}$$

3.4 Utility planning

For determination of the *target utility TU* the following formula is used within the model:

$$TU = U_{BS} + CS \tag{8}$$

with U_{BS} the utility of the own bid, and the concession step CS determined by

$$CS = \beta(1 - \mu/U_{BS})(U_{BO} - U_{BS}) \tag{9}$$

In the above formula, U_{BS} and U_{BO} represent the utilities of the agent's own (or self-bid), respectively the utility of the other's bid, in the immediately previous negotiation step. The factor $U_{BO} - U_{BS}$ expresses the current *utility gap*. It is important to note that in Eq. 9 both utilities are computed with respect to the agent's own utility function. Thus, if the utility of the self (own) bid is the actual, true utility value, the utility of the opponent's bid is only an approximation, since the true utility of the other remains unknown. This illustrates a key aspect of negotiations with incomplete preference information (i.e. where the utility function of the opponent is unknown): the perceived remaining utility gap between the own and other's offer may be very from the perspectives of the two parties. A concession made by one agent (in the way of decreasing his own utility function) may be perceived as no concession at all (even a retraction) by the other party. Similarly, an agent can make no concession (or a very small concession), but the opponent perceives this concession as very considerable. It is precisely this type of concessions that must be achieved in multi-attribute negotiations, since it means the new offer is closer to the Pareto-efficient frontier. Such concessions can only be achieved by finding suitable concession trade-off between the issues.

The factor $(1 - \mu/U_{BS})$ expresses that the concession step will decrease to 0 if the U_{BS} approximates the minimal utility μ. This ensures $U_{BS} \geq \mu$. The factor β stands for the *negotiation speed*. The minimal utility is taken as $\mu = 1 - \gamma$ with γ the *concession factor*, expressing a measure in how far concessions can be made.

The concession factor plays the role of a reservation utility (i.e. the amount of utility, compared to the maximum expected utility (which in our model is 1), which the agent is willing to forgo or concede in a negotiation. The rationale for this is that, in human negotiations, often the players do not know their true or absolute cut-off value (minimal utility), but start working from a higher value. If it turns out that with that somewhat higher minimal value, they cannot close the deal, humans tend to adjust this minimal value closer to their true cut-off value. This aspect is modeled by us through the concession factor: if the automated negotiation does not lead to a deal, the human user (if one is present) is asked if he wants to lower the cut-off value by adjusting the concession factor.

Determination of the target utility can also address the ease and financial aspect separately indicated by E or F added to the parameters, while subscripts *BS* and *BO* continue to denote the self (own) bid and the other's bid, respectively. For the ease aspect the following formula is used:

$$TEU = EU_{BS} + ECS \tag{10}$$

$$ECS = \beta_E(1 - \mu_E/EU_{BS})(EU_{BO} - EU_{BS}) \tag{11}$$

In this formula β_E is the negotiation speed factor for the ease part, and μ_E is the minimal ease utility. It is important to point out that this is the *overall* target ease utility for the entire bid. The concession is not evenly distributed across the attributes: in some attributes larger concessions are made, while in others little or no concessions occur. The method for computing concession in each attribute is presented in Sect. 3.5. For the financial aspect the target utility is:

$$TFU = FU_{BS} + FCS \tag{12}$$

$$FCS = \beta_F(1 - \mu_F/FU_{BS})(FU_{BO} - FU_{BS}) \tag{13}$$

The speed factors β_E for ease and β_F for financial parts are based on the negotiation speed factor β and the financial rationality factor ρ as follows

$$\beta_E = (1 - \rho)\beta \quad \beta_F = \rho\beta \tag{14}$$

The *minimal ease utility* is taken as $\mu_E = 1 - \gamma$. The *minimal financial utility* is taken as $\mu_F = \varepsilon/\delta$ where ε is the *minimal financial margin*. The explanation is as follows. If the minimal margin is achieved, then the price *minP* is

$$minP = \varepsilon b + b + a_B \tag{15}$$

Given *minP*, the minimal acceptable financial utility can be calculated as follows:

$$\mu_F = (minP - b - a_B)/\delta b = \varepsilon b/\delta b = \varepsilon/\delta \tag{16}$$

For example, if $\delta = .2$ (20%) and $\varepsilon = .1$ (10%), then $\mu_F = .5$, i.e. the dealer is not willing to sell with a financial utility lower than half of its maximal financial utility (based on the maximal margin); a financial utility of 0 means selling against the cost price, i.e. no margin at all, a financial utility of 1 means selling with a margin of 20% on the cost price. As shown in Sect. 3.3, this is a natural choice in the business domain we consider.

3.5 Attribute planning

The Attribute Planning process uses as input the target utility and determines as output the configuration for the next (own) bid in the following two main steps:

- First, within the component Target Evaluation Determination, for each attribute a target evaluation is determined.

- Next, given these target evaluations per attribute, within the component Configuration Determination, a configuration for the next bid is determined.

We identify two possible negotiation strategy choices, corresponding to two possible designs of the internal designs of the Attribute Planning sub-component:

- Attribute Planning without explicit Opponent Modeling
- Attribute Planning with Opponent Modeling.

In the classification above, by "opponent modeling" it is meant that the seller agent maintains an estimation of the preference weights that the buyer assigns to different attributes and takes them into account in computing the target evaluations.

The case without opponent modeling is suitable for perfectly closed negotiations, when no information about the weight of any of the attributes is shared between parties and when the negotiations are too short to reliably deduce ("guess") this information.

The second option, in which the seller does use opponent modeling was introduced [27] in order to improve the Pareto-efficiency of the reached agreements (this is presented in Sect. 3.9). In this case, the Opponent's (i.e. negotiation partner's) preference weights are taken into account in computing the target evaluation for different attributes and determining the configuration of the next bid. We implemented and experimentally compared the two cases: one in which opponent modeling is not attempted and one with opponent modeling (i.e. guessing his preferences between bid history). We show that, for the experimental setting we considered, the model which uses guessing succeeds in improving the Pareto-efficiency of the reached agreements over the model in which no guessing is attempted. This is especially true for the case there is at least some partial preference information disclosed before the negotiation and the trace of the negotiation so far is reveals the opponent's preferences for some of the values.

3.6 Attribute planning without opponent modeling

If the Attribute Planning is done without trying to estimate the opponent's (i.e. negotiation partner's) preference weights, then the process takes only 2 main steps:

- Determining, for each attribute, a target evaluation the attribute should have in the next bid (task assigned to the Target Evaluation Determination sub-component)
- Determining the actual configuration of the next bid, by assigning values to the discrete attributes (task assigned to the Configuration Determination sub-component)

The internal composition of this component in this case is given in Fig. 5. Following, we discuss the operations performed in each of the shown components.

3.7 Target evaluation determination

Target evaluations per attribute TE_j are determined in the model in two steps. First a *basic target evaluation* per attribute BTE_j is determined in such a way that $\Sigma_j\, w_j\, BTE_j = TU$. Then the *target evaluations* TE_j are combinations of the BTE_j with the evaluations of the attributes in the bid of the negotiation partner. The basic target evaluation per attribute BTE_j is determined according to the following format:

$$BTE_j = E_{BS,j} + (\alpha_j/N)(TU - U_{BS}) \tag{17}$$

Here the α_j represent the concession speed per attribute and N is a normalization factor. Factor N is defined as the weighted sum of the α's with the relative importance factors as weights: $N = \Sigma_j w_j \alpha_j$. Due to this normalization factor, the utility determined as a combination of the target evaluations leads to exactly the target utility:

$$\begin{aligned} \Sigma_j w_j BTE_j &= \Sigma_j w_j (E_{BS,j} + (\alpha_j/N)(TU - U_{BS})) \\ &= \Sigma_j w_j E_{BS,j} + \Sigma_j w_j (\alpha_j/N)(TU - U_{BS}) \\ &= U_{BS} + 1/N\, \Sigma_j w_j \alpha_j (TU - U_{BS}) \\ &= U_{BS} + 1/N * N * (TU - U_{BS}) \\ &= TU \end{aligned}$$

The choice for the α's is made as: $\alpha_j = (1 - w_j)\,(1 - E_{BS,j})$. The first factor expresses the influence of the user's own importance factors and is chosen as an inverse to the agent's preference weights (similar to the choice made in Ref. [1]); the second factor takes care that the target evaluation values remain scaled in the interval between 0 and 1. Besides the influence on the target attribute evaluations as described, also a concession to the opponent's attribute evaluations is made. This depends on the configuration tolerance τ, as follows:

$$TE_j = (1 - \tau)BTE_j + \tau E_{BO,j} \tag{18}$$

If the configuration tolerance is 0, then only the user's importance factors are taken into account. If the configuration tolerance is 1, then with respect to the configuration maximal concession to the negotiation partner is made. Note that in this case, although the preferences expressed in the partner's last bid are taken into account, the concession rate only takes into account the agent's own weights — and does not try to predict anything about the preferences of her opponent. More crucially, the configuration tolerance τ is the same for all attributes, which leads to a relatively uniform concession rate.

When information about the opponent's preferences is available, the concession towards the values from the other's previous bids can be made more asymmetric for different attributes, as shown in Sect. 3.9.

3.8 Configuration determination

To determine a configuration for the next bid the following three steps are made.

- First, for each attribute, given the target evaluation, attribute values are determined with an evaluation that is as close as possible to the target evaluation value.
- Next, a partial configuration (price attribute not yet filled) is determined based on these closest values.
- Finally, to complete the configuration for the next bid, the price attribute value is determined.

The partial configuration is selected from the closest attribute values. If more than one choice with closest value is possible, then, if it is among the options, the value in the opponent's bid is chosen, otherwise the choice is made in a random manner. The partial configuration is completed by determining the price attribute value in such a manner that the overall target utility is achieved.

Within the Dealer Representative (Seller) agent a simple possibility would be to take the target financial utility as the aim to be exactly achieved. However, due to the discrete values of the accessory attributes, the ease utility can probably not be exactly achieved. The choice has been made that this difference is compensated in the financial utility. For example, if the ease utility of the partial configuration is lower than the target ease utility, then the seller agent aims at a financial utility which is (in proportion) higher than the target financial utility.

First the ease utility of the partial configuration is determined. Next the financial utility that has to be achieved (AFU) is determined, as the (weighted) difference between overall target utility and the realized ease utility:

$$AFU = TU - (1-\rho)U_{P,E}/\rho \tag{19}$$

where $U_{P,E}$ is the ease utility of the partial configuration P.

Finally, the price attribute value is determined, as the sum of all costs and the fraction of the maximum margin given by the financial utility aimed for:

$$price = b + a_P + AFU\delta b \tag{20}$$

3.9 Attribute planning with opponent modeling

In this section, we consider the situation when, in order to make better directed concessions, in planning the target evaluation for each attribute we take into account not only the own preference weight of the agent, but also the weight of the opponent. The adapted process composition is presented in Fig. 6. If the opponent is not willing to reveal her preference weight for some (or maybe all) attributes, an estimation of these weights is computed in component Estimation of Opponent's Parameters. The role of the component Guess Coefficients is to analyze the way the opponent is bidding and to provide some extra information to be used for estimating these private preference weights. In the following we discuss these components in separate sections.

3.10 Target evaluation planning

This component outputs a target evaluation for each attribute in the next bid, based on the bid target utility value. The target attribute evaluation is determined in two steps. First a basic target attribute evaluation for each attribute is computed according to Eq. 17 from Sect. 3.7, for the case when no opponent modeling is attempted. The basic target evaluation (c.f. Eq. 17) considers only the own attribute preference weights of the agent. It represents a concession, compared to the utility offered in the own, immediately previous bid, but it can be a poorly directed concession. Using only this value works in general, but it can lead to sub-optimal results, since the preferences of the other are not considered in any way when making concessions. To improve on this, the following solution was implemented. For each attribute $j \in A$ (where A denotes the set of all attributes) a Preference Difference Coefficient δ_j is computed as:

$$\delta_j = (W_{other,j} - W_{own,j})/(W_{other,j} + W_{own,j}) \tag{21}$$

This coefficient (scaled between -1 and 1) expresses how different the preferences of the two parties for each attribute are. Positive values for δ_j denote a stronger preference of the negotiation partner for attribute j, while negative values denote a stronger own preference for this attribute. Note that the weights of the other for all the attributes in the above formula are not generally known quantities. The opponent may reveal (some of) his weights — but for all attributes whose weights are not known, these will need to be heuristically approximated. This will be discussed in Sect. 3.11. The concession to be made in each attribute $j \in A$ depends on a parameter called *configuration tolerance*, denoted as $\tau_j \in [0, 1]$. The tolerance parameter is chosen to be attribute-specific, in order to better differentiate the amount of concessions between attributes. Therefore, for each attribute $j \in A$, the configuration tolerance depends on the preference difference coefficient of that attribute, according to the following formula:

$$\tau_j = \tau_{gen} * (1 + \delta_j) \tag{22}$$

Here the parameter τ_{gen} represents the general tolerance, used by the agent for all attributes *j*. The general tolerance is always chosen between 0 and .5 and also gives a measure of how fast the agent is willing to make concessions. Values closer to 0 will denote an agent who is less willing to make concessions, while values closer to .5 will denote an agent who is interested to reach a deal quickly.

Finally, the target evaluation for each attribute j is computed. This is done by taking into account both the basic target attribute evaluation (as described above) and a concession to the attribute evaluation from the previous bid of negotiation partner, as follows:

$$TE_j = (1 - \tau_j)BTE_j + \tau_j E_{BO,j} \tag{23}$$

Here BTE_j is the basic attribute evaluation for attribute *j* and $E_{BO,j}$ is the evaluation for attribute *j* from the opponent's previous bid. From the above formula, one can see that values of the configuration tolerance τ_j close to 0 signify that mostly the user's own importance factors are taken into account, while values close to 1 shows that maximum possible concession is made towards the other's value. And since τ_j depends directly on δ_j, it is the difference in preference for each attribute that determines how much concession should be made.

By choosing $\tau_{gen} \in$[0, .5], we ensure that always $\tau_j \in$ [0,1] (the attribute-specific tolerance is always scaled between 0 and 1), and thus the target evaluation per each attribute is a linear combination between the own valuation and the opponent's valuation for each attribute. This can be easily shown as follows.

From Eq. 21, we see that the attribute specific delta δ_j takes a minimum value of -1 (when $W_{own,j} = 1$ and $W_{,other,j} = 0$). This means the agent assigns a maximal value to attribute *j* (its weight is equal to the maximum of 1), while the opponent is indifferent to the value in this attribute. Consequently, in this case, from Eq. 22 we get $\tau_j = \tau_{gen} * (1 - 1) = 0$. Applying Eq. 23 will mean that $TE_j = BTE_{j,}$ i.e. no concession will be made toward the valuation of the opponent in this attribute.

The opposite case in Eq. 21 occurs when $W_{own,j} = 0$ and $W_{other,j} = 1$, and thus the delta value for this attribute is maximal $\delta_j = 1$ (i.e. the agent is indifferent to an attribute, but the opponent assigns a maximal value to it). In this case, the attribute specific tolerance is $\tau_j = 2 * \tau_{gen}$. Since $\tau_{gen} \in$ [0, .5], this means that τ_j can take a maximum value of 1, which after applying Eq. 23 results in $TE_j = E_{BO,j}$, i.e. a complete concession is made towards the value of the opponent in this attribute. In general, however, τ_{gen} is chosen as less than .5, such that the resulting concession is more gradual. This is actually an advantage, because the evaluation of the opponent is an approximation, so a too fast concession may mean giving up too quickly in the agent's own utility, without significantly increasing the opponent's utility. A value of τ_{gen} above .5 is never justified, since it may result in τ_j for some attributes being greater than 1 (i.e. more concession is made in those attributes than is even asked by the negotiation partner), which is not realistic.

Because, in our model both the sum of the agent's own weights and sum of the opponent's weights are always scaled to 1, the above mechanism leads to a situation where greater concessions in some attributes (more important to the opponent) will always be balanced by smaller concessions in other attributes (more important to the agent itself). Such an asymmetric concession system allows both negotiating parties to reach greater utility quicker.

In this component we took the opponent's preference weights for each attribute as known. However, if the other is not willing to share his weights for some (or all) attributes, then they will need to be estimated. This is discussed in the following sections.

3.11 Estimation of opponent's parameters

This component determines, for those attributes for which the opponent was not willing to reveal his preference weights, an estimation of those weights.

We denote by A_{known} the set of attributes for which the opponent was willing to reveal his importance weights in the beginning of the negotiation and by $A_{unknown}$ the attributes whose preference weights are kept private. Since all preference weights are normalized (see Sect. 3.1), the sum of weights for the private attributes is computed as:

$$\Sigma_{j \in A_{unknown}} W_j = 1 - \Sigma_{k \in A_{known}} W_k \tag{24}$$

For attributes with private weights, the remaining weight $\Sigma_{j \in A_{unknown}} W_j$ has to be divided between them. For this purpose we assign a parameter called the Remaining Weight Distribution Coefficient R_j to each attribute $j \in A_{unknown}$.

These attributes can be further classified into two sub-sets:

- Attributes for which a reliable guess about the preference of the opponent can be made based on her previous bids (we denote this class by $A(G)$). These attributes will be assigned a coefficient R_j in the component Guess Coefficients (as described in Sect. 3.12).
- Attributes for which no reliable information about the preference weights of the opponent can be made from his previous bids (denoted by $A(NG)$). These attributes are assigned a default value $R_j = 2$, which is empirically chosen between the values for attributes for which there is an indication they are important to the opponent (from her past bids) and those attributes which are less important to her (see Table 1). After establishing the value of this parameter, the estimation of the actual weight is computed as follows:

$$W_j = (R_j / \Sigma_{k \in A_{unkown}} R_k) * \Sigma_{k \in A_{known}} W_k \tag{25}$$

The rationale for this choice is to insure that the sum of all weights remains (unknown and known) remains normalized to 1, without the weight distribution coefficients having necessarily add up to 1. This is useful, because in this context, the values of the weight distribution coefficients are empirically chosen. In our domain and problem setting, the values that produced the best results are reported in Table 6 below. However, if applying our negotiation model to other domains and/or problem settings (e.g. increasing the number of attributes, having a higher/lower number of discrete values etc.) other values of this coefficient may work best. This allows the user of the system to experiment with different heuristic values, while Formula (22) and (23) nevertheless ensure that the weights estimated remain normalized to 1.

Table 1 Action decision table

Action	Repetition	Utility gap	Configuration match
Next round	No		No
Discuss concession factor	Yes	Yes	No
Discuss configuration tolerance	Yes	No	No
Report success		No	Yes
Report failure	Yes		No

It is also possible that no reliable information can be obtained from the opponent's past bids for any of the attributes. Then all distribution coefficients will be equal and applying the above formula results in equal distribution of the remaining weight between private attributes, formally expressed as: $A_{unknown} = A_{(NG)} => \forall j, k \in A_{unknown}, W_j = W_k$.

3.12 Guess coefficients

This component analyses the opponent's bids and, for those attributes for which a trend is reliable detected, returns a value for the remaining weigh distribution coefficient.

In the current model the explicit assumption used in guessing (for the Seller's side only) is that, everything else being equal, a human Buyer would prefer a better quality item to a poorer quality one. Otherwise stated there exists a (partial) ordering of the attribute values such as: *evaluation(good) > evaluation(fairly good) > evaluation (standard) > evaluation(meager) > evaluation(none)*. We define the Attribute Value Distance AVT_j for each attribute $j \in A$ as the distance between values for an attribute in two successive bids, on an ordinal scale. For example, given the above ordering, the distance between good and fairly good is 1, while the difference between good and standard is 2. It is important to show that this attribute value distance does NOT depend on the exact values the opponent assigns to these labels — since in the current model this information is private (not disclosed to the other). After running a considerable number of experiments we observed that such a simple ordering information can lead to a reasonably good heuristic. Partial ordering information is usually sufficient to make a good prediction about the opponent's preferences in the negotiation (i.e. if this distance is known only for some labels, this is enough).

Next we need a mapping of the detected concession distances to the remaining weight distribution coefficients introduced in Sect. 3.4 (see Table 2). The values for the above coefficients were determined experimentally as follows: first between each two different labels (representing quality levels) an initial value was computed by subtracting their distance value from 4 (the maximum distance). Then the parameters were adjusted to provide a best linear fit for the results over a large number of tests. This mapping is domain-specific, meaning it led to good results in the tests we performed, for our specific application. Therefore this parameter mapping may need to be adjusted by the system designer/agent owner upon application in other domains, but our model is generic enough to allow this flexibility.

Another issue to be discussed is how many successive bids in the negotiation trace need to be analyzed in order to make a prediction for R_j. From our empirical tests we observed that in most cases it is sufficient to adjust the R_j parameter based only on

Table 2 Remaining weight distribution coefficients assigned to attribute value distances for attrib. j∈ A(G)

Attribute value distance(j)	$R(j)$
0	6
1	4
2	3
3	1
4	.5

the first 3 bids. This can be explained by the fact that our model, being cooperative, agreement over the attributes with discrete values occurs in the first rounds of the negotiation — and usually the last rounds can be characterized as "haggling" over the only continuous attribute, the price.

4 Implementation and experimental validation

The model introduced in Sect. 2 was implemented in the DESIRE agent platform, build at the Vrije Universiteit, Amsterdam [4]. However, the conceptual negotiation model presented in Sect. 2 is platform independent. In fact, after the publication of our original research [15,17], a negotiation model conceptually very similar to ours, but implemented in the more commercial Java Aglets platform was presented in Ref. [29].

This section first discusses the experimental set-up used in testing the model presented in Sect. 2. Following, a full example trace is presented for the implementation. Finally the aggregate experimental results (for different test parameters) are presented.

4.1 Experimental set-up

In order to test the robustness of the above model, we considered the following dimensions:

- The number of attribute weights revealed
- Whether guessing is used or not
- The choice for the attribute importance factors
- The evaluations for the attribute value levels

Besides these dimensions, the model contains a number of general parameters, see Table 3 for an overview. The space of possible parameter instantiations in this case is very large. Therefore, after some preliminary testing, we choose to fix the parameters at the *overall bid level* at a set of values. These values were chosen such that they provide both stable behavior of the system and assure a termination property — i.e. in most instances lead to the conclusion of the negotiation in a relatively small number of negotiation steps (around 10–20 for most tests).

The aim of our test model is to test whether the agents are able to explore efficiently the space of possible deals in this number of steps.

The importance factors assigned to different attributes are presented in Table 4. Note that these were also chosen to provide a sufficient cover of the space of possible preferences. The values presented in Table 4 are raw importance factors, which are then normalized to add up to 1, using the formula presented in Sect. 3.

Table 3 Values assigned to different parameters which concern the negotiation strategy by the Seller and Buyer at the overall bid level

Negotiation parameter	BR	DR
Negotiation speed β	.5	.4
Impatience factor π	4	4
Configuration gap size in price ν	250	200
Utility gap size ω	.02	.02
Concession factor γ	.5	.9
Configuration tolerance τ	.5	.9
Financial rationality factor ρ	Not applicable	.5
Minimal financial margin ε	Not applicable	.1
Maximal financial margin δ	Not applicable	.3

Table 4 Importance factors used for Buyer/Seller, for different levels of preference asymmetry

	Tow hedge	Airco	Extra speakers	CD player	Price
Fully asymmetric	90/15	90/15	15/90	15/90	300/300
Partially asymmetric	53/53	90/15	15/90	53/53	300/300
Fully symmetric	53/33	53/53	53/53	53/53	300/300

Table 5 Value levels Good/Fairly Good/Standard/Meager/None for each of the 4 attributes

	Profile 1	Profile 2
Buyer	100/85/70/30/0	100/70/50/35/0
Seller	30/65/80/65/100	30/50/70/85/100

Next, we should check that these results hold for different possible value configurations. Again the search space here is very large, so we must restrict our attention to a few profiles combinations, which are shown in Table 5.

In our tests, we assume a business model in which the Seller prefers to sell the car for a standard price — and not have to install extra accessories, but he is willing to do so in order to sell it. On the buyer side, because in our model the values for the attributes represent quality labels, the distances between utilities assigned to each label can be interpreted as how "quality conscious" or selective that buyer is. For example, by looking at Table 5, a buyer of profile 2 is more selective than a buyer of profile 1, because his utility for "fairly good" and "standard" qualities drops quicker, when compared to the optimal quality level "good". Other choices are possible, but in order to properly test the model the choice for the values must be asymmetrical – meaning the two parties would like different values for each attribute. Otherwise the parties quickly agree on the configuration (since their interests are convergent) and the negotiation reduces to haggling about the price.

4.2 An example negotiation trace

In this section, we illustrate the model presented in Sect. 2 through an example. Here we take the negotiation between a Buyer and Seller with totally asymmetric preferences (see Table 4), where the only information revealed between parties is the normalized weight of 1 attribute (Tow hedge). For accessories, for both Buyer and Seller, profile 1 is used (see Table 5). For this example, we use the perspective of the Seller, which in our case is the party using guessing. For reasons of space, we can illustrate only a small part of the full mathematical model, but we hope it is enough for the reader to understand the rationale behind some of our design choices.

The attribute *Tow Hedge* has the following normalized preference weights (see Table 3):

$$W_{BUYER,TowHedge} = 90/(90+90+15+15+300) = .1764$$
$$W_{SELLER,TowHedge} = 15/(90+90+15+15+300) = .0294$$

From the perspective of the Seller the preference Difference Coefficient for *Tow Hedge* will be: $\delta_{TowHedge} = (W_{Buyer,TH} - W_{Seller,TH})/(W_{Buyer,TH} + W_{Seller,TH}) = (.1764 - .0294)/(.1764 + .0294) = .714$.

A positive value close to 1 (as shown in 2.3), indicates this the attribute is more important to the other party (the Buyer). As the general tolerance (for the Seller side) in this case is $\tau_{gen} = .3$, the attribute specific tolerance will be $\tau_{TowHedge} = \tau_{gen} * (1 + \delta_{TowHedge}) = .3 * (1 + .714) = .514$. Since $\tau_{TowHedge} > \tau_{gen}$, a larger concession than average will be made towards the Buyer's requested value in this attribute. This can be seen in Table 6 as a large concession, in the first round from "none" to "fairly good". Next we exemplify the guessing of the opponent's weights discussed in Sects. 2.4 and 2.5. We do this only after the first two rounds from the opponent's bids, though the mechanism is the same for subsequent rounds.

The Value Distances and Remaining Weight Distribution Coefficients for the unknown attributes are (see Table 5 for the Buyer's first 2 bids and 1 for the coefficient mapping):
$VD(Airco) = VD(good, standard) = 2 => R_{Airco} = 3$
$VD(CD\_player) =s VD(Speakers) = VD(good,meager) = 3$
$=> R_{CD\_player} = R_{Speakers} = 1.$

Since $\Sigma_{j \in A_{unknown}} W_j = 1 - (15 + 300)/510 = .235$, the estimated weights are: $W_{AIRCO} = 3/(1+1+3) * .235 = .141$, $W_{CD\_PLAYER} = 1/(1+1+3) * .235 = .047$

In this case, the estimations produced by the guessing are not far from the true (non-revealed) values of the Buyer: .176 for *Airco* and .0294 for *CD player*.

Table 6 provides the complete trace of this negotiation from the perspective of the Buyer, while Table 7 does the same from that of the Seller. The vertical columns show the bids made by the two parties in successive rounds.

Figure 7 provides a visualization of the negotiation progress in the joint utility space (as automatically produced by the implementation in our software environment). For clarity, only the first 3 bids of the Buyer (marked with a "B") and the first 2 of the

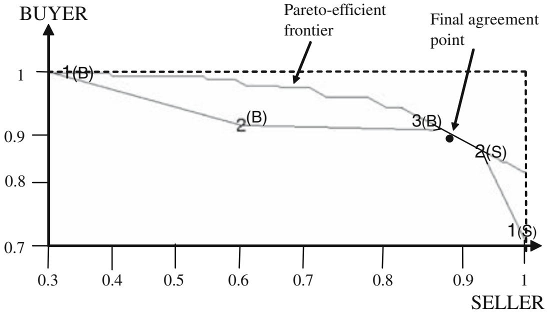
Table 6 The negotiation trace: Buyer's perspective

Buyer	1	2	3	4	5	Closing
Bids						
Price	18,000	17,450	17,968	18,047	18,083	18,083
Tow hedge	Good	Fairly good	Fairly good	Fairly good	Fairly good	Fairly good
Airco	Good	Standard	Standard	Standard	Standard	Standard
Speakers	Good	Meager	None	None	None	None
CD player	Good	Meager	None	None	None	None
Utilities						
Own bid	1	.9203	.9130	.9094	.9068	.9068
Seller's bid	.7407	.8782	.8830	.8864	.8889	.8889

Table 7 The negotiation trace: Seller's perspective

Seller	Round 1	2	3	4	5	accept:5
Bids						
Price	16,900	18,468	18,404	18,359	18,325	18,083
Tow hedge	None	Fairly good	Fairly good	Fairly good	Fairly good	Fairly good
Airco	None	Standard	Standard	Standard	Standard	Standard
Speakers	None	None	None	None	None	None
CD player	None	None	None	None	None	None
Utilities						
Own bid	1	.9378	.9296	.9238	.9195	.8884
Buyer's bid	.3167	.5932	.8737	.8838	.8884	.8884

Fig. 7 Utility space corresponding to the example trace from Tables 6 and 7



Seller (marked "S") are shown. The remaining offers all lie close together in the straight line between point 3(B) of the buyer and point 2(S) of the seller. This is an interesting effect, which we have seen in a number of negotiation traces: after establishing mutually agreeable values for the discrete-value attributes, the agents seem to "walk" the Pareto-efficient frontier towards each other's bid. This corresponds to the haggling about the price from rounds 3–5 in Tables 6 and 7.

The effect is particularly remarkable, since one must keep in mind that neither one of the agents knows exactly where the Pareto-frontier lies. They cannot compute this information because they only have partial knowledge of the opponents utility function (in our experiments, the position of the frontier was computed after the fact using the full information, but only as a benchmark to measure performance — i.e. without giving this information to the agents). This effect is surprising, since agents in our model only model the opponent preferences, they do not try to actually predict where the frontier lies. Nevertheless, we observed that if opponent modeling is performed efficiently, the agents "discover" the Pareto-frontier implicitly, using their approximate opponent models.

4.3 Comparing traces from the same test set

We define a test set as the set of all negotiation traces which share the same Pareto-efficient frontier and therefore whose outcomes are directly comparable. Between the negotiations in the same set, the preferences of the two parties are the same: the only difference is the amount of information shared and their willingness to use guessing.

Test sets are distinguished from each other by two (sets of) parameters:

- The level of asymmetry in the attribute importance factors (see Table 4)
- The distances between the value labels (cf. Table 5)

Comparing the efficiency of outcomes, both within the same test set and between test sets, is important in our setting, since, as is shown here, the efficiency of the outcomes which can be reached depends on how asymmetric the preferences of the parties are. In Sect. 4.3.1, we first discuss the results from a test set with maximum preference asymmetry (this setting also corresponds to the example in Sect. 4.2 above). In Sect. 4.3.2 we discuss the results from two related test sets, but where preferences are more asymmetric (all the results from Sects. 4.3.1 and 4.3.2 refer to the values profile 1, for both Buyer and Seller). Finally, in Sect. 4.4 we present the aggregate results for all test sets, across all value distance settings and preference symmetry profiles considered.

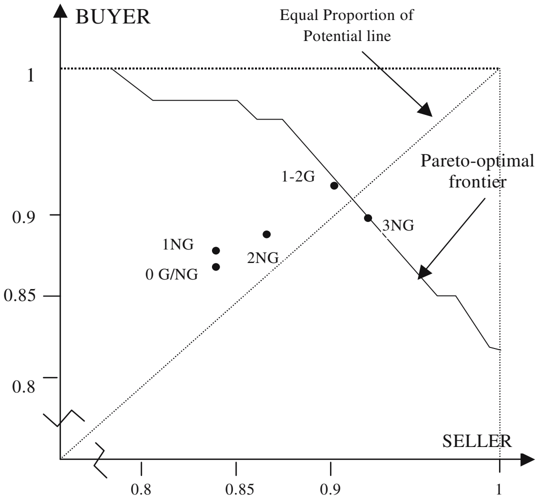
4.3.1 Test set with maximal preference asymmetry

Table 8 shows the final outcomes of negotiations involving a Buyer and Seller with asymmetric preferences and value profiles 1, while in Fig. 8 these outcomes are plotted w.r.t the Pareto-optimal frontier. The notation is: 1.3 denotes the number of attributes shared and NG/G denotes whether guessing is used or not. The Pareto frontier in Fig. 8 is the same as in Fig. 7, just scaled between different values. In fact, the outcome reached in Fig. 7 appears as point 1G in Fig. 8. The irregular, non-convex shape of the Pareto-efficient frontier (computed according to [26]) is typical for real-life domains, where some attributes take discrete values and only some are continuous.

Table 8 Distribution of final outcomes for the negotiations between a Buyer with a stronger preference for the attributes *Drawing Hook* and *Airco*, and a *Seller* with a stronger preference for *CD player* and *Extra Speakers*

Experimental setting	Price	Tow hedge	Airco	Extra speakers	CD player	Buyer utility	Seller utility
Closed negotiation (with or without guessing)	19,018	Standard	Standard	Standard	Standard	.865185	.84177
One attribute weight revealed, without guessing	19,178	Fairly good	Standard	Standard	Standard	.86822	.84054
One attribute weight revealed, with guessing	18,325	Fairly good	Standard	None	None	.906815	.91946
Two attribute weights revealed, without guessing	18,790	Fairly good	Standard	None	Standard	.882741	.87211
Two attribute weights revealed, with guessing	18,325	Fairly good	Standard	None	None	.906815	.91946
Three attribute weights revealed (with or without guessing)	18,428	Fairly good	Fairly good	None	None	.914	.90166

Fig. 8 Outcomes for negotiations between a Buyer and Seller with asymmetric preference weights



From the above test set we can already make some observations. First, more attribute weights shared improves the outcome, so the mechanism is able to make efficient use of incomplete preference information. In Table 6, this is illustrated by the fact that, as more attribute weights are shared, a better match is obtained between the preferences of the parties, with each one obtaining its preferred values in the more important attributes. Second observation is that the guessing heuristic may improve the outcome, sometimes considerably. In the trace presented for 1 or 2 attribute weights shared guessing helps bring the outcome very close to the Pareto-efficient frontier. For 0 attribute weights shared (i.e. perfectly closed negotiation), in this particular test set guessing does not help much (however there are test sets where it does). In the 3 attribute weights shared case the outcome without guessing is already Pareto-efficient. Note, however, that this case is not equivalent to fully open negotiation, because the evaluations for the values assigned to each quality level are still not revealed between parties.

4.3.2 Test sets with symmetric and partially symmetric preferences

In this section we illustrate two other test cases, related to the one above, because the distances between values for the buyer and seller are still 1 (cf. Table 5). Figure 9 plots the outcomes from two test sets: the first one in which the preference weights of both parties are the same across all 4 discrete-valued attributes, the second one in which only two attributes have equal preference weight, the other two having asymmetrical weights (see Table 4 for the exact values).

From Fig. 9, it can be observed that, in fact, for more symmetric preferences revealing more information and/or using guessing does not make too much difference (the tables with the exact outcomes reached are not given here for lack of space, but they point to the same conclusion). In fact, for the case with completely symmetric preferences (Fig. 9(a)) we can see that all outcomes actually overlap. In this special case (equal weights across all attributes), the negotiation actually becomes a zero-sum game, since there are no mutually beneficial trade-offs between attributes, so the best that can be achieved is to settle on the middle of the range value.

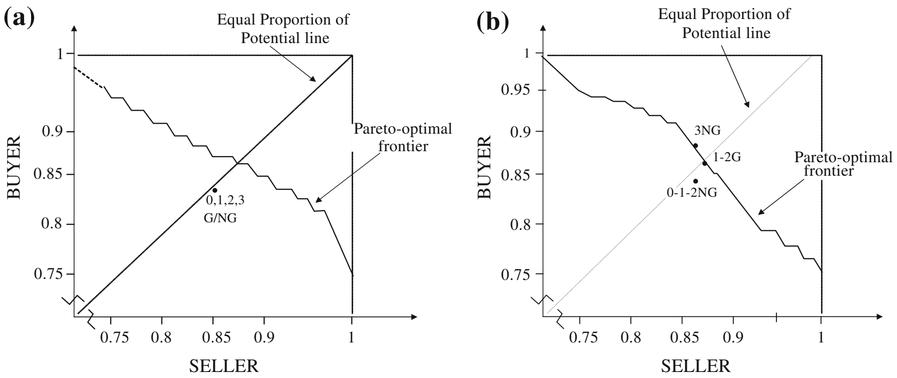


Fig. 9 (**a**) Outcomes of the negotiation between a buyer and seller with completely symmetric preferences (i.e. all attributes have equal weights for both parties). (**b**) Outcomes of the negotiation between a buyer and seller with partially symmetric preferences (i.e. two attributes have equal weights, two are asymmetrical — c.f. Table 4)

4.4 Comparing results from all test sets

As shown in Sect. 4.1, 96 negotiation traces were generated in order to test the validity of our model. Due to space limitation we cannot present in the same detail all the experimental results (the interested reader is asked to consult [27]). In Fig. 10 we show the average utilities across all tested profiles, grouped by the level of asymmetry in preference weight between parties. Within each group, from left to right the level

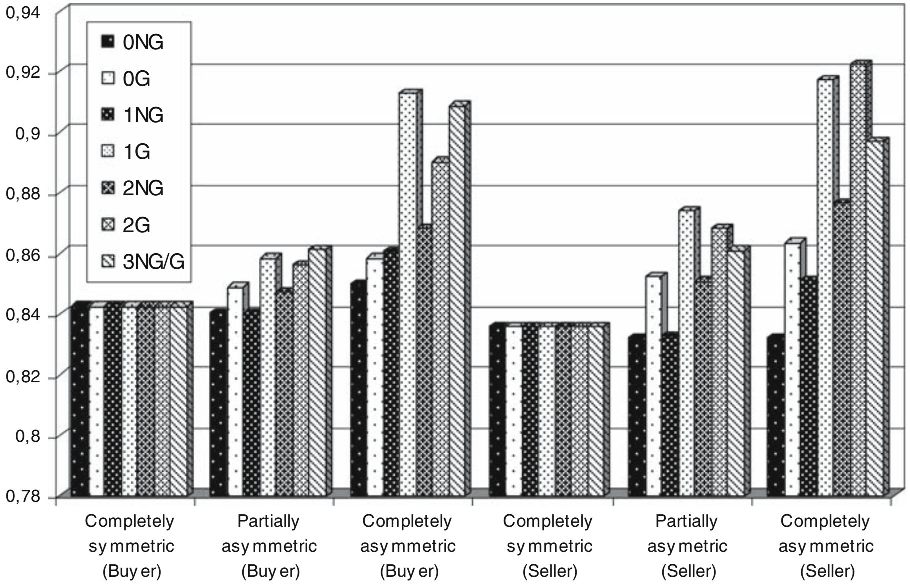


Fig. 10 Average utilities for all profiles tested, for different cases of preference asymmetry and openness

of openness is varied from no attributes revealed and no guessing used to 3 attributes weights revealed.

Based on Fig. 10, we can see that our observations from Sect. 4.3 generalize across profiles: both sharing more information and guessing improves the utility (on average). It can be seen that the more asymmetrical the preferences of the two parties are, the greater the scope for potential gains that can be obtained either by sharing more information or using the guessing heuristic. For example, for all profile combinations tested in the perfectly symmetrical preferences case, the outcome always had a 0% improvement, either from sharing more preference weight information or by using the guessing heuristic. By contrast in the partially symmetric preferences improvements were of the order of 3–4%, which went up to around 10% for asymmetric preference weights. This effect can be explained by the fact that our mechanism exploits precisely this preference asymmetry in order to increase the efficiency of the joint outcome for both parties.

Another important conclusion is that, if the negotiation speed (see Sect. 4.1) is set the same for both parties, the outcomes will always lie relatively close to the equal proportion of potential line, regardless of the guessing/openness model used. Otherwise stated, the overall concession for the bid level are similar, even though for each attribute may differ widely. This ensures that, if the negotiation outcome lies on, or close to, the Pareto-efficient frontier, it will also be relatively close to the Kalai–Smorodinsky bargaining outcome. This may be important, since some sources (e.g. [26]) consider closeness to this point as a measure of "fairness" of the negotiation outcome.

4.5 Human–computer experiments

The results reported in this paper refer only the automated negotiation case, i.e. the case when both Buyer and Seller are represented in the negotiation by automated software agents. The user (or owner of the agent) in this case, only needs to input its preference parameters (i.e. parameters corresponding to its utility function in the multi-attribute space, c.f. Sect. 3.3), and the software agent computes the bids/counter-offers. We have, however, also considered the case when humans propose their own bids against software agents — and we tested the system using an experimental economics type of approach (also put forward in Ref. [6]). The full results are outside the scope of this paper, and they have been comprehensively reported elsewhere [2,3]. Here we provide only a brief synopsis of the results, interested readers being asked to refer to the above references for the full model.

The negotiation model has been tested extensively involving more than 70 students. The students were asked to create their own profile. That profile was used in all settings: they were asked to negotiate against our software agents and against other students; the same profiles were used to perform agent–agent negotiations, see Refs. [2,3]. The results show that comparing agent–agent negotiations with human-human negotiations with the same profiles the agents usually outperformed the humans. However, some intuitive leaps performed by humans, are lacking in the agent system. Out of all our participants (more than 70), some 4 were bound on and successful in manipulating the agent when against it. It is important to note, though, that all other players just took the task of negotiation seriously without attempting to find the limitations of the agent. Interestingly, when comparing human–agent negotiation with agent–agent negotiations, the humans got somewhat better results than the agents, and both did clearly better than in the corresponding human–human setting. In future, the negotiation agents using guessing will tested in the same way.

Finally, we should mention that our mechanism was designed to prevent obvious ways of cheating, like over-stating attribute preference weights. This is because each agent scales the sum of the preference weights declared by the opponent to 1. So an agent has no incentive to over-state his preferences for any attribute, since this may lead to the opponent making smaller or no concessions in other attributes. Furthermore, a system was added by which an agent stops negotiating when it detects insufficient concessions from the other in several successive bids, which should prevent situations where one party makes all the concessions. However, in designing any distributed mechanism, the problem incentive compatibility remains a challenging one [8] and formal proofs of the truth-revelation properties of the proposed negotiation protocol was outside the scope of this work.

5 Discussion

In this section, we provide an overview existing work on negotiation, and, by comparison, we discuss different aspects from our own model.

In Gutman and Maes [13] a number of criteria and benefits are discussed of some different approaches to negotiation. For example, in the competitive negotiation system Kasbah three negotiation strategies are mentioned: anxious (linear increase of bids over time), cool-headed (quadratic), and frugal (exponential). In the model presented here, these strategies can be used to determine the negotiation speed. Another important issue discussed in Ref. [13] is the argument for cooperative negotiation that merchants often care less about profit on any given transaction and care more about long-term profitability, which implies customer satisfaction and long-term customer relationships. That argument supports the importance of the following factors in our model for negotiation: configuration tolerance (consumer satisfaction), concession factor (profit), minimal financial margin (profit), and financial rationality (profit). Furthermore, the remark that cooperative negotiation is a win–win type of negotiation is supported by our model in that consumers and providers both have an extensive multi-attribute profile (importance factors, evaluation descriptions) that influence the outcome of the negotiation aiming to satisfy both parties.

The argumentation approach to negotiation (see for e.g. [24] for an overview) allows the agents to exchange not only bids, but also arguments that influence other agents' beliefs and goals, which, it is claimed, allows more flexibility. Some issues which are usually left open in such approaches are: how do the agents' mental states relate to their utilities and if (or how) can the efficiency of such negotiations be measured from game-theoretic perspective.

A main difference of our work to the work described in Benn et al. '99 [1] is that in our approach it is possible to specify heuristics both for the overall utilities and for separate attributes (with their values and evaluations). In their approach no overall view is made; a compensation matrix is used to compensate a concession in one attribute by other attributes. In our approach it is possible to decide about the overall concession (in terms of the overall utility) in a negotiation step, independent of specific concessions for separate attributes. Moreover, in their approach a neural (Hopfield) network is used to find the compensations for attributes by an approximation process.

In contrast, our approach uses explicit knowledge to determine the attributes of a new bid, which makes it more transparent and better explainable.

Fatima et al. [12], present a model for bilateral multi-attribute negotiation, where attributes are negotiated sequentially. The issue studied is the optimal agenda for such a negotiation under both incomplete information and time constraints. However, a central mediator is used and the issues all have continuous values. The effect of time on the negotiation equilibrium is the main feature studied, from both a game-theoretic and empirical perspective. In earlier research [11] a slightly different model is proposed, but the focus of the research is on time constraints and the effect of deadlines on the agents' strategies.

Lai et al. [20] propose a multi-issue negotiation strategy over continuous-valued issues that involves choosing an offer on the iso-utility (or "indifference") curve that is closest to the best counter-offer made by the opponent in the previous negotiation round. Luo et al. [23] look at the multi-issue negotiation problem in semi-cooperative environments, and propose a solution based on fuzzy constraint reasoning. Brzostowski and Kowalczyk [5] study a related problem to multi-issue negotiation: that of selection of negotiation partners, based on possibilistic methods.

Dang and Huhns [9] study agent-mediated, multi-issue negotiation, in the context of internet-based service provision. Hemaissia et al. [14] propose a multi-issue negotiation model for another setting: that of distributed crisis management.

The line of research of Klein et al. [18] is continued in Ref. [19] for domains in which issue dependencies influence the overall utility of a bid. Klein at al. [18] show that there is no guaranteed efficient method that an agent can use to negotiate over multiple issues, even if the agent tries to guess the opponent's profile. They propose to use a mediator who uses a computationally expensive evolutionary algorithm that can solve non-linear optimization tasks of high dimensionality.

We argue that the most related work to ours is represented by Faratin et al. [10]. Like Faratin et al., we start from the perspective of distributed negotiation, which eliminates the need of a central planner. As in Ref. [10], we also take the heuristic approach and we model agents that are able to jointly explore the space of possible outcomes with a limited (incomplete) information assumption. In Ref. [10], this is done through a trade-off mechanism, in which the agent selects the value of its next offer based on a similarity degree with previous bids of the opponent. In our design, we do no explicitly model trade-offs, yet the same effect is achieved through the asymmetric concessions mechanism. An advantage of our model over [10] is that we allow agents to take into account not only their own weights, but also those of the opponent in order to compute the next bid. In this way agents may exchange partial preference information for those attributes for which their owners feel this does not violate their privacy. Also the initial domain information for the attributes with discrete ("qualitative") evaluation is different. In Ref. [10], this consists of fuzzy value labels, while in our model it is a partial ordering of attribute weights.

In more recent work, Coehoorn and Jennings [7] extend the model proposed by Faratin et al. [10], with a method to learn opponent preference weights, based on kernel density estimation.

Somefun et al.'03 [30] propose a system for bilateral negotiation over bundles of information goods. A novel derivative-follower Pareto search strategy is proposed, in order to search for optimal bundles in the space of the agent's ISO-utility lines. In related work, Lai et al. '06 [21] propose a system for multi-issue negotiation which is based on minimizing the perceived distance between the opponent's previous offer and the agent's own ISO-utility line.

Finally, a paper really related to our work is Shakshuki and Abu-Driaz [29]. They propose a peer-to-peer, agent-mediated e-commerce system, that uses multi-issue negotiation as one of its core components. The negotiation model employed by [29] is virtually the same as the one presented in this paper — and was inspired by it, after the initial, conference publication of our work [15–17,27]. Indirectly, this paper provides another validation of the conceptual negotiation model presented here, since it represents an independent re-implementation of this model in a different domain and software platform (in Ref. [29], the Java Aglets platform was used instead of DESIRE).

6 Conclusions

The paper introduces a component-based generic agent architecture for integrative multi-attribute negotiation. An application of the model is described in a prototype system for negotiation about cars, developed in cooperation with, among others, Dutch Telecom KPN. The model is robust against some obvious ways of cheating, and stalling, but the dedicated opponent can manipulate our agents.

In the approach the privacy of both parties is protected. However, it also provides both parties to reveal a limited amount of information, i.e. the weight of one or more attributes. The model contains a mechanism in which agents use any amount of incomplete preference information revealed by the negotiation partner in order to improve the efficiency of the reached agreements. Furthermore, a guessing mechanism is introduced in this paper, that is shown to improve the outcome of the negotiation even further. In this guessing heuristic the agent uses the history of the opponent's bids to predict his preferences. Experimental evaluation shows that the combination of these two strategies leads to agreement points close to or on the Pareto-efficient frontier. A point that inspires further research is our finding that if both agents use the same value for the negotiation speed parameter, the outcome is also close to the Kalai–Smorodinsky point. The negotiation model has also been extensively tested against student negotiators. The results show that in general the use of agents improves the negotiation outcome, be it in agent–agent negotiations or human–agent negotiations.

Therefore, the main original contributions of this paper are:

- The component-based generic agent architecture for integrative multi-attribute negotiation,
- A negotiation strategy that has proved itself in experiments with humans,
- A guessing strategy that further improves the outcome of the negotiation,
- Revealing only a limited amount of preference information to each other leads to significant joint gains in the outcome.

The negotiation model described in this paper has been designed and automatically implemented using the DESIRE software environment, developed at Vrije University in Amsterdam — and is part of a larger Brokering framework, which also includes other aspects of B2C electronic commerce, such as product search and matching (as outlined in Sect. 2). However the applicability of the conceptual negotiation model presented in this paper is not strictly limited to our implementation framework and could be re-used in further research. For example, after the publication of our original model in Refs. [16,17], an agent-based negotiation framework, conceptually very similar to ours, but implemented in the Java Aglets platform has been presented in Ref. [29].

Acknowledgements The authors wish to thank Lourens van der Meij for his assistance with the software implementation, Tibor Bosse for the human experiments data and the anonymous reviewers for their detailed and useful comments.

References

1. Benn, W., Goerlitz, O., & Neubert, R. (1999). Enabling integrative negotiations by adaptive software agents. In *Cooperative Information Agents III*, LNAI Vol. 1652 (pp. 335–346). Springer-Verlag.
2. Bosse, T., & Jonker, C. M. (2005). Human vs. computer behaviour in multi-issue negotiation. In T. Ito, H. Hattori, T. Matsuo, & M. Zhang (Eds.), Proceedings of the first international workshop on rational, robust, and secure negotiations in multi-agent systems, RRS'05, pp. 10–25.
3. Bosse, T., Jonker, C. M., & Treur, J. (2004). Experiments in human multi-issue negotiation: Analysis and support. In N. R. Jennings, C. Sierra, L. Sonenberg, & M. Tambe, (Eds.), Proceedings of the third international joint conference on autonomous agents and multi-agent systems, AAMAS'04. ACM Press, pp. 672–679. Extended version: Bosse, T., Jonker, C. M., van der Meij, L. & Treur, J. (2007). Automated formal analysis of human multi-issue negotiation processes. *Multi-agent and Grid Systems Journal* (to appear).
4. Brazier, F. M. T., Jonker, C. M., & Treur, J. (2000). Compositional design and reuse of a generic agent model. *Applied Artificial Intelligence Journal, 14*, 491–538.
5. Brzostowski, J., & Kowalczyk, R. (2006). Experimental evaluation of posibilistic mechanism for negotiation partners selection. In *Proceedings of second international workshop on rational, robust, and secure negotiations in multi-agent systems*, Hakodate, Japan. Springer Lecture Notes in Computational Intelligence (to appear).
6. Byde, A., & Chen, K.-Y. AutONA: A System for Automated Multiple 1–1 Negotiation. *Fourth ACM Conference on Electronic Commerce*, (2003) pp. 198–199.
7. Coehoorn, R. M., & Jennings, R. M. (2004). Learning an opponent's preferences to make effective multi-issue negotiation tradeoffs. In *Proceedings of sixth international conference on e-commerce, delft,* ACM Press, pp. 59–68.
8. Dash R. K., Parkes D. C., & Jennings N. R. (2003). Computational mechanism design: A call to arms. *IEEE Intelligent Systems, 18*(6), 40–47.
9. Dang, J., & Huhns, M. H. (2006). Concurrent multiple-issue negotiation for internet-based services. *IEEE Internet Computing*, 10(6), 42–49.
10. Faratin, P., Sierra, C., & Jennings, N. (2003). Using similarity criteria to make issue trade-offs in automated negotiations. *Journal of Artificial Intelligence, 142*(2), 205–237.
11. Fatima, S., Wooldridge, M., & Jennings, N. R. (2002). Optimal negotiation strategies for agents with incomplete information. In *Intelligent Agents VIII*, LNAI, Vol. 2333, (pp. 377–392). March 2002, Springer-Verlag.
12. Fatima, S. S., Wooldridge, M., & Jennings, N. R. (2003). Optimal agendas for multi-issue negotiation. *Second international conference on autonomous agents and multiagent systems (AAMAS-03)*, Melbourne, July 2003, pp. 129–136.
13. Gutman, R., & Maes, P. (1998). Cooperative vs. competitive multi-agent negotiation in retail electronic commerce. In *Proceedings of the second international workshop on cooperative information agents (CIA'98)*, NAI Vol. 1435, (pp. 135–148) Springer Verlag, Paris.
14. Hemaissia, M., Seghrouchni, A. E. F., Labreuche, C., & Mattioli, J. (2006). Cooperation-based multilateral multi-issue negotiation for crisis management. In *Proceedings of second international workshop on rational, robust, and secure negotiations in multi-agent systems*, Hakodate, Japan. Springer LNCI (to appear).
15. Jonker, C., & Robu, V. (2004). Automated multi-attribute negotiation with efficient use of incomplete preference information. In *Proceedings of third international conference on autonomous agents and multi-agent systems (AAMAS-04), New York*, pp. 1056–1064, ACM Press.
16. Jonker, C. M., & Treur, J. (1999). A re-usable broker agent architecture with dynamic maintenance capabilities. In O. Etzioni, J. P. Mueller, & J. Bradshaw (Eds.), *Proceedings of the third*

annual conference on autonomous agents, Agents'99. ACM Press, 1999, pp. 376–377. Extended version: Jonker, C. M., & Treur, J. (2002). Compositional design and maintenance of broker agents. In L. C. Jain, Z. Chen, & N. Ichalkaranje (Eds.), *Intelligent Agent and their Applications*, (pp. 149–171). Springer-Verlag.

17. Jonker, C. M., & Treur, J. (2001). An agent architecture for multi–attribute negotiation. In B. Nebel (Ed.), *Proceedings of the seventeenth international joint conference on AI, IJCAI'01*, 2001, pp. 1195–1201.
18. Klein, M., Faratin, P., Sayama, H., & Bar-Yam, Y. (2002). Negotiating complex contracts. In *Autonomous agents and multi-agent systems*. Bologna:AAAI Press.
19. Klein, M., Faratin, P., Sayama, H., & Bar-Yam, Y. (2003). Protocols for negotiating complex contracts. *IEEE Intelligent Systems Journal, Special Issue Agents & Markets, 18*(6), 32–38.
20. Lai, G., Li, C., Sycara, K., & Giampapa, J. (2004). Literature review on multi-attribute negotiations, Technical Report CMU-RI-TR-04–66, Carnegie Mellon University, Robotics Institute.
21. Lai, G., Sycara, K., & Li, C. (2006). A decentralized model for multi-attribute negotiations with incomplete information and general utility functions. In *Proceedings of second international workshop on rational, robust, and secure negotiations in multi-agent systems*, Hakodate, Japan. Springer LNCI (to appear).
22. Lewicki, R., Saunders, D., & Minton, J. (1997). *Essentials of negotiation*. Irwin Press.
23. Luo, X., Shadbolt, N., Leung H., Lee J. H., & Jennings, N. R. (2003). A fuzzy constraint based model for bilateral multi-issue negotiations in semi-competitive environments. *Artificial Intelligence Journal, 142*(1–2), 53–102.
24. Rahwan, I., Ramchurn, S. D., Jennings, N. R., McBurney, P., Parsons, S., & Sonenberg L. (2004). Argumentation-based negotiation. *Knowledge Engineering Review* (to appear).
25. Raiffa, H. (1982). *The art and science of negotiation*. Cambridge, Mass.:Harvard University Press.
26. Raiffa, H. (1996). *Lectures on negotiation analysis*. *PON Books, Harvard Law School*.
27. Robu, V. (2003). Improving the efficiency of cooperative negotiations in electronic environments with incomplete information. Master Thesis, Vrije Univ, Amst.
28. Rosenschein, J. S., & Zlotkin, G. (1994). *Rules of Encounter*. MIT Press, Cambridge, Massachussets.
29. Shakshuki, E., & Abu-Draz, S. (2005). Agent-mediated e-commerce system. In *Proceedings of the nineteenth international conference on advanced networking and applications (AINA'05)*, Vol. 2, pp. 739–744, IEEE Computer Society Press.
30. Somefun K., Gerding E., Bohte S., & La Poutré, H. (2003). Automated negotiation and bundling of information goods. In *Agent-mediated electronic commerce V*, Melbourne, Australia, July.
31. Strobel, M. (2000). Effects of electronic markets on negotiation processes. *IBM Research Technical Report, Zurich*.