Chapter 2
Alternative Social Welfare Definitions for Multiparty Negotiation Protocols

Enrique de la Hoz, Miguel Angel Lopez-Carmona, Mark Klein, and Ivan Marsa-Maestre

Abstract Multiagent negotiation protocols, understood as a group decision making process, try to reach an agreement among all the negotiating agents. Traditionally, this agreement is an unanimous agreement. This consensus as unanimity may be quite difficult to achieve in practice or even undesirable in some situations. We propose a framework to incorporate alternate consensus definitions to multiagent negotiations in terms of utility sharing among the agents. The consensus definition is enforced by a mediator, which implements a linguistic-expressed mediation rule based on *Ordered Weighted Averaging Operators* (OWA). In each step of the mediation process, agents send offers to the mediator. To avoid zones of no agreement, the mediator applies *Hierarchical Clustering* (HC) to the offers to form group of agents. Then, the mediator computes a social contract, taking into account the desired consensus and the distance from an ideal consensus. The social contract is submitted as a feedback to the agents that explore locally the negotiation space using of a variation of the *Generalized Pattern Search* (GPS) non-linear optimization technique to generate new offers that into account the social contract. Finally, We show how these mechanisms are able to reach agreements according to different consensus policies while avoiding zones of no agreement.

Keywords Coalition formation coordination • Negotiation • Teamwork

E. de la Hoz (✉) • M.A. Lopez-Carmona • I. Marsa-Maestre
Computer Engineering Department, Universidad de Alcala, Alcala de Henares, Spain
e-mail: enrique.delahoz@uah.es; miguelangel.lopez@uah.es; ivan.marsa@uah.es

M. Klein
Center for Collective Intelligence, MIT Sloan School of Management, Massachusetts Institute of Technology, Cambridge, MA, USA
e-mail: m_klein@mit.edu

I. Marsa-Maestre et al. (eds.), *Novel Insights in Agent-based Complex Automated Negotiation*, Studies in Computational Intelligence 535,
DOI 10.1007/978-4-431-54758-7_2,

2.1 Introduction

Multi-attribute negotiation may be seen as an interaction between two or more agents with the goal of reaching an agreement about a range of issues which usually involves solving a conflict of interests between the agents. Although, this should constitute an incentive for them to cooperate and search for possible joint gains, self-interested agents often fail to reach consensus or end up with inefficient agreements.

On one hand, self-interested agents would like to reach an agreement that is as favourable to them as possible. On the hands, final decision is jointly made and needs to be agreed to by both the agents. As a result of this, negotiation agents have to consider how much they could gain individually if they cooperate and in which way of cooperation they could gain more, or at least receive a fair deal. Negotiation protocols should include techniques for dealing fairly with rational agents that also are able to lead them to mutually beneficial agreements. Because of this, a fundamental objective of any negotiation protocol should be to optimize some type of social welfare measurement [1]. There are many different social welfare measurements like the sum or product of utilities, the min utility, etc. [2–4].

In spite of that, social welfare has not been taken into account as an integral part of the negotiation process. There are some works that incorporate a social welfare criterion within the search process, though. In [5], the mediator generate jointly preferred proposals for agreements. By iteratively moving along jointly improving directions from the tentative agreements produced by the method, negotiating parties can achieve joint gains and finally reach a Pareto-optimal agreement. The procedure is repeated until no further joint improvements can be found. In [6] a mediator assists decision makers in finding Pareto-optimal solutions. Decision makers have to indicate their most preferred points on different sets of linear constraints. The method can be used to generate either one Pareto-optimal solution dominating the status quo solution of the negotiation or an approximation to the Pareto frontier. In [7], a non-biased mediator agent searches for the compromise directions based on a E-DD (Equal Directional Derivative) approach and supports negotiation agents in reaching an agreement. At each stage of negotiation, the mediator searches for the compromise direction based on a new E-DD (Equal Directional Derivative) approach and computes the new tentative agreement.

These solutions have some important restrictions. First, the utility functions have to be derivable and quasiconcave. Second, the absolute value of gradient is not considered, so that the marginal utility obtained by the agents may not be fair. Third, the protocol is prone to untruthful revelations of information to bias the direction generated by the mediator. Finally, the protocols do not allow to specify the desired consensus on the final agreement.

The traditional or strict notion of consensus in multi-agent negotiation protocols, commonly known as unanimity, assumes that consensus exists only if all agents agree on a contract. Unanimous agreements may be quite difficult or even impossible to achieve in practice and, in some cases, undesirable. Alternate definitions of consensus, as soft-consensus [8] have been proposed that consider different degrees of partial agreement among agents to decide about the existence of consensus on an contract. Consensus measures based on soft consensus are more can be used to reflect linguistic expressions of mediation rules by using linguistic quantifiers.

In this work, we propose a framework to incorporate the type of consensus desired to reach and agreement as an integral part of multiparty negotiation protocols. We propose HCPMF, a Hierarchical Consensus Policy based Mediation Framework for Multi-Agent Negotiation. HCPMF implements a mediation protocol that is based on the *Generalized Pattern Search* (GPS) non-linear optimization technique [9], the use of *Ordered Weighted Averaging* (OWA) operators [10,11], and the use of *Hierarchical Clustering* (HC) [12]. GPS is used by the agents to perform local exploration of the negotiation space, HC lets the mediator to form clusters of agents to avoid zones of no agreement, and OWA operators are used to apply the consensus policies, which are captured using linguistic quantifiers. Globally, HCMPF allows to efficiently search for agreements following predefined consensus policies, which may take the form of linguistic expressions. The protocol is designed to minimize the revelation of private information. Agents only propagate offers to the mediator, not their preferences for the offers. Furthermore, agents' offers need not to be known by their opponents.

Next section presents the basic operation of the negotiation protocol. Then we present a variation of the GPS algorithm to perform local exploration of the negotiation space and the mediation mechanisms. Two last sections describe the experimental evaluation and present our conclusions.

2.2 The Negotiation Protocol

We shall assume a set of n agents $A = \{A_1, \ldots, A_n\}$ and a finite set of issues $X = \{x_1, \ldots, x_m\}$ in a continuous or discrete domain. A *contract* is a vector $\mathbf{x} = \{x'_1, \ldots, x'_m\}$ defined by an instance of issue values. Each agent A_i has a real mapping $U_i : X \to \mathbb{R}$ function that associates with each contract $\mathbf{x}$ a value $U_i(\mathbf{x})$ that gives the payoff the agent assigns to a contract. The preference function can be described as any mapping function between the negotiation space contracts and the set of real numbers, and it can be non-monotonic and non-differentiable. The aim of the agents will be to reach an agreement on a contract $\mathbf{x}$ maximizing their individual payoff while minimizing the revelation of private information.

2.2.1 Basic Operation of the Negotiation Protocol

The basic protocol of the negotiation process is as follows:

1. Each agent sends the mediator an ***initial contract offer***. This offer may be the result of a local utility maximization process, or a contract generated at random.

2. Based on the received offers, the mediator applies the ***HC algorithm*** to form *clusters of agents*. The cluster with the highest number of agents is selected.
3. The mediator applies the ***OWA operator*** to the offers in the selected cluster to obtain a *feedback contract*. The OWA operator synthesizes the consensus policy to apply. Finally, the mediator verifies if the deadline has been reached. If so, negotiation ends with an agreement on the feedback contract. Otherwise, go to step 4.
4. The mediator computes the ***group distance***, which is a distance estimate to the current feedback contract from the offers in the cluster. If the group distance is below a threshold the negotiation ends with an agreement on the feedback contract. Otherwise go to step 5.
5. The mediator proposes the ***feedback contract*** to the agents.
6. Each agent performs a *local exploration* of the negotiation space using ***GPS*** to generate a *new offer*. The agent's exploration considers the feedback contract and utility. Go to step 2.

In the next section we will present the GPS non-linear optimization algorithm that will be used by agents to explore the contract space.

2.3 Agents' Local Exploration (GPS)

Each agent privately explores the negotiation space using a variation of the GPS [9] non-linear optimization algorithm. GPS belongs to the family of *Direct Search Based* optimization algorithms. Formally, the optimization problem can be defined as max $f(x)$, where $f : \mathbb{R}^m \to \mathbb{R}$, $x \in \mathbb{R}^m$ represents the evaluation of the contracts in terms of distance, utility or both. At an iteration k of the protocol, we have an iterate $x(k) \in \mathbb{R}^m$ and a step-length parameter $\Delta_k > 0$. We will use the notation $x^{+o}(k)$ to designate the mesh at round k plus the current point $x(k)$ (see Fig. 2.1.

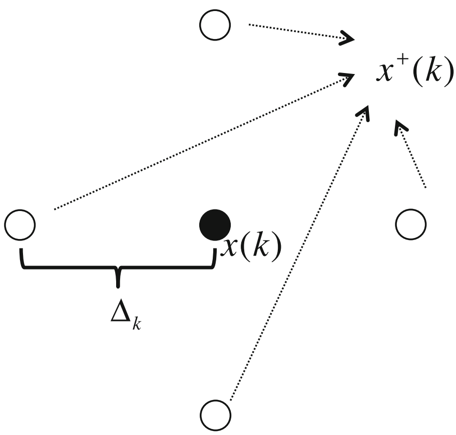

Fig. 2.1 An illustration of a mesh for $m = 2$ at round k. The reference point is $x(k)$

This set of points or mesh is an instance of what we call a *pattern*. One important feature of pattern search that plays a significant role in a global convergence analysis is that we do not need to have an estimate of the derivative of f at $x(k)$ so long as included in the search is a sufficient set of directions to form a positive spanning set for the cone of feasible directions, which in the unconstrained case is all of $\mathbb{R}^m$. The set e is defined by the number of independent variables in the objective function m and the positive standard basis set. A commonly used positive basis is the maximal basis, with $2m$ vectors. For example, if there are two independent variables in the optimization problem, the default for a $2m$ positive basis consists of the following pattern vectors: $e_1 = \{1, 0\}$, $e_2 = \{0, 1\}$ and $-e_1 = \{-1, 0\}$, $-e_2 = \{0, -1\}$.

The exploration begins at the first negotiation round with the generation of an initial random contract (*reference contract*) and a set of contracts (*mesh*) around the reference contract at a predefined distance. The reference contract will be the offer to be submitted to the mediator that will compute a feedback contract, taking into account the reference contracts received from all the agents and will send it back. Then, we successively evaluate the points in the *mesh* $x^+(k) = x(k) \pm \triangle_k e_j$, $j \in \{1, \ldots, m\}$, in terms both of utility and of distance to the *feedback contract* provided by the mediator (evaluations will be better for higher utilities and shorter distances). This set of points or mesh is an instance of what we call a *pattern*. If one or more contracts $x'(k)$ in $x^+(k)$ in the mesh improve the reference contract both in utility and distance, the contract with the highest improvement becomes the current reference contract ($x(k+1) = x'(k)$), and a new mesh is generated increasing by a factor of 2 the step-length factor, $\triangle_{k+1} = 2 \cdot \triangle_k$. Otherwise, the agent has to decide if to behave as a utility maximizer, considering only the contracts' utility in the evaluation, or as a utility conceder, considering only the distance to the feedback contract. We model the agents' attitude using a random variable. In any of these cases, if the improvement is in the mesh, that is, at least there exists a $x'(k)$ that improves $x(k)$ either in terms of utility or distance but not in both, the contract with the highest improvement ($x'(k)$) becomes the current reference contract, and a new mesh $x^+(k+1)$ is generated increasing by a factor of 2 the current step-length factor,$\triangle_{k+1} = 2 \cdot \triangle_k$. If there is no point $x'(k)$ in the mesh $x^+(k)$ that improves the current reference contract $x(k)$, the reference contract remains the same ($x(k+1) = x(k)$) and a new mesh $x^+(k+1)$ is generated at half the current step-length, $\triangle_{k+1} = 0.5 \cdot \triangle_k$.

2.4 The Mediation Mechanisms

The goal of the mediation process is to provide a useful feedback to the agents to guide the joint exploration of the negotiation space implementing the desired consensus while avoiding zones of no agreement. This feedback is represented by the *feedback contract* or *social contract*. The mediation process takes into account not only the utility of the offers but also their distance to the social contract.

This mediation process, at any round k, can be described as follows:

1. The HC algorithm is applied to the agents' offers $O_k = \{\mathbf{o}_{k1}, \ldots, \mathbf{o}_{kn}\}$ in order to form clusters of agents
2. For the contracts in the highest sized cluster $O_{kc} = \{\mathbf{o}_{kc1}, \ldots, \mathbf{o}_{kcl}\}$, the centroid $\mathbf{c_k}$, the distances $D_{kc} = \{d_{kc1}, \ldots, d_{kcl}\}$ from the contracts to the centroid and the set of direction vectors $R_{kc} = \{\mathbf{r_{kc1}}, \ldots, \mathbf{r_{kcn}}\}$ from the centroid to the contracts are computed.
3. The sets O_{kc}, D_{kc} and R_{kc} are ordered from lower to higher distances (distances in D_{kc}). The set D_{kc} is normalized in the range $[min(D_{kc}), 0]$, $min(D_{kc})$ representing the lower distance and 0 the higher distance.
4. The OWA operator that represents the desired consensus policy will be applied to these values in order to obtain the feedback contract.
5. To assess the convergence to a solution the mediator also computes the *group distance* as the OWA-weighted distances to the feedback contract.

Next we will go into detail in each of the steps performed by the mediator at each round k. First, we will describe the clustering mechanism, second, the procedure to obtain the feedback contract, which includes the description of the aggregation procedures used to model the consensus policy, and finally, the computation of the group distance.

2.4.1 Forming Clusters of Agents (HC)

Here we look at the process whereby the mediator obtains the highest sized cluster of agents at each negotiation round. We have used an Hierarchical Clustering (HC) algorithm [12] to perform this task. HC groups data over a variety of scales by creating a cluster tree or dendrogram. The tree is not a single set of clusters, but rather a multilevel hierarchy, where clusters at one level are joined as clusters at the next level. This allows us to decide the level or scale of clustering that is most appropriate at each step of the negotiation process.

In our case, we assume that the mediator has defined an upper bounded number of rounds as a deadline. This number of rounds nr in divided into stages. Thus, we have *ns* stages with *nr/ns* rounds per stage. At each stage, a predefined scale of clustering is applied. In our case, the mediator applies the scales of clustering in descending order. It means that as negotiation progresses the clustering process is more prone to generate clusters. The rationale behind this is that we first try to reach agreements with as many agents as possible, and if we are not able to reach a global agreement we progressively form smaller groups where the negotiation process is focused on agents with closer preferences. In order to vary the scale of clustering a cutoff level is varied which specifies the level at which the hierarchy of clusters is cut.

2.4.2 Computing the Feedback Contract

Our point of departure here is the collection of l contracts corresponding to the highest sized cluster. For this set of contracts, the mediator computes the centroid $\mathbf{c}_k$, the distances D_{kc} and the set of direction vectors R_{kc}. The mediator's objective is to obtain a feedback contract that better represents a predefined consensus policy.

If the consensus policy is to keep as many agents satisfied as possible, under complete uncertainty, the mediator could propose the centroid as a compromise solution. On the other hand, if the consensus policy is to have for instance at least one agent satisfied with a high utility, the feedback contract should be biased towards the contracts closer to the centroid. To develop these ideas we use the quantifier guided aggregation technique which is implemented through the use of OWA operators. This mechanism is a refinement with respect to the clustering mechanisms. While the purpose of HC is to avoid zones of no agreement, the aim of using OWA operators is to apply a predefined consensus policy.

2.4.2.1 OWA Operators

Our goal is to elicit a function M which takes $\mathbf{c}_k$, D_{kc} and R_{kc} in order to obtain a feedback contract following a consensus policy. The form of M is called the *mediation rule*, it describes the process of combining the individual agents' preferences. The form of M can be used to reflect a desired mediation imperative or *consensus policy* for aggregating the preferences of the individual agents to get the feedback contract. The most widespread consensus policy found in the automated negotiation literature suggests using as an aggregation imperative a desire to satisfy *all* the agents. We propose to use application dependent mediation rules to manage the negotiation processes. The idea is to use a *quantifier guided aggregation*, which allows a natural language expression of the quantity of agents that need to agree on an acceptable solution. As we shall see, the OWA operators [11] will provide a tool to model this kind of softer mediation rule.

We define two types of aggregation operators, scalar and vectorial.

Definition 2.1. An scalar OWA operator of dimension l is a mapping $M : S^l \to G, (S, G \in [0, 1])$ such that, $M(S_1, \ldots, S_l) = \sum_{t=1}^{l} w_t b_t$, where b_t is the tth largest element of the aggregates $\{S_1, \ldots, S_l\}$ and the w_j are weights such that $w_t \in [0, 1]$ and $\sum_{t=1}^{l} w_t = 1$

Definition 2.2. An vectorial OWA operator of dimension l is a mapping $M : S^l \to G, (S, G \in \mathbb{R}^m)$, such that, $M(S_1, \ldots, S_l) = \sum_{t=1}^{l} w_t b_t$, where b_t is the tth largest element of the vectorial aggregates $\{S_1, \ldots, S_l\}$ and the w_j are weights such that $w_t \in [0, 1]$ and $\sum_{t=1}^{l} w_t = 1$

It can be shown [11] shows that OWA aggregation has the following properties:

1. Commutativity: The indexing of the arguments is irrelevant
2. Monotonicity: If $S_i \geq \hat{S}_i$ for all i then $M(S_i, \ldots, S_n) \geq M(\hat{S}_i, \ldots, \hat{S}_n)$
3. Idempotency: $M(S, \ldots, S) = S$
4. Boundedness: $Max_i[S_i] \geq M(S_i, \ldots, S_n) \geq Min_i[S_i]$

In the OWA aggregation the weights are not directly associated with a particular argument but with the ordered position of the arguments. If *ind* is an index function such that *ind*(t) is the index of the tth largest argument, then we can express M as:

$$M(S_1, \ldots, S_l) = \sum_{t=1}^{l} w_t S_{ind(t)} \tag{2.1}$$

The form of the aggregation is dependent upon the associated weighting vector. We have a number of special cases of weighting vectors. The vector W^* defined such that $w_1 = 1$ and $w_t = 0$ for all $t \neq 1$ gives us the aggregation $Max_i[S_i]$. Thus, it provides the largest possible aggregation. The vector W_* defined such that $w_l = 1$ and $w_t = 0$ for all $t \neq n$ gives the aggregation $Min_i[S_i]$. An interesting family of OWA operators are the E-Z OWA operators [13]. There are two families. In the first family we have $w_t = 1/q$ for $t = 1$ to q, and $w_t = 0$ for $t = q + 1$ to l. Here we are taking the average of the q largest arguments. The other family defines $w_t = 0$ for $t = 1$ to q, and $w_t = \frac{1}{l-q}$ for $t = q + 1$ to l. We can see that this operator can provide a softening of the original min and max mediation rules by modifying q.

2.4.2.2 Quantifier Guided Aggregation

There are several approaches to perform OWA weights identification [14], including methods based on maximum entropy, on previous observations of decision makers performance [15]. In this work, we will derive OWA weights from linguistic quantifiers [11]. Our final objective is to define consensus policies in the form of a linguistic agenda. For example, the mediator should make decisions regarding the generation of the feedback contract following mediation rules like "*Most* agents must be satisfied by the contract", "*at least* α agents must be satisfied by the contract", "*many* agents must be satisfied", ...

The previous examples are examples of *quantifier guided aggregations*, which are aligned with the notion of soft-consensus, which we discussed earlier. Linguistic quantifiers [16] can be used to semantically express aggregation policies and actually capture Kacprzyk's notion of soft consensus.

OWA weights identification based on linguistic quantifiers is possible thanks to fuzzy set theory. There are two types of linguistic quantifiers: absolute and relative [16]. Any relative linguistic quantifier can be expressed as a fuzzy subset Q of the unit interval $I = [0, 1]$ [10]. In this representation for any proportion

$y \in I$, $Q(y)$ indicates the degree to which y satisfies the concept expressed by the term Q. Relative linguistic quantifiers can be classified into three categories: Regular Increasing Monotone (RIM) quantifier, Regular Decreasing Monotone (RDM) quantifier and Regular UniModal (RUM) quantifier[11]. RIM quantifiers allow us to model the notion of soft consensus[17]. Formally, these quantifiers are characterized in the following way:

1. $Q(0) = 0$
2. $Q(1) = 1$
3. $Q(x) \geq Q(y)$ if $x > y$.

Examples of this kind of quantifier are *all*, *most*, *many*, *at least* α. According to this representation, the quantifier *all* can be represented by Q_* where $Q_*(1) = 1$ and $Q_*(x) = 0$ for all $x \neq 1$, and *any* which is defined as $Q^*(0) = 0$ and $Q^*(x) = 1$ for all $x \neq 0$. It has been shown [11] that the OWA weights can be parametrized using this kind of functions.

Under the quantifier guided mediation approach a group mediation protocol is expressed in terms of a linguistic quantifier Q indicating the proportion of agents whose agreement if necessary for a solution to be acceptable. The basic form of the mediation rule in this approach is "Q agents must be satisfied by the contract", where Q is a quantifier. The formal procedure used to implement the mediation rule is as follows:

1. Use Q to generate a set of OWA weights $W = w_1, \ldots, w_l$.
2. Use the weights W to calculate the feedback contract.

The procedure used for generating the weights from the quantifier is to divide the unit interval into n equally spaced intervals and then to compute the length of the mapped intervals using Q

$$w_t = Q\left(\frac{t}{l}\right) - Q\left(\frac{t-1}{l}\right) \text{ for } t = 1, \ldots, l\,. \tag{2.2}$$

In Fig. 2.2 we show an example of a linguistic quantifier and illustrate the process of determining the weights from the quantifier. The weights depend on the number of agents as well as the form of Q. In Fig. 2.3 we show the functional form for the quantifiers *all*, *any*, Q_*, Q^*, *at least* α *percent*, *linear quantifier*, *piecewise* Q_{Z_β} and *piecewise* Q_{Z_α}.

The quantifiers *all*, *any* and *at least* α describe the consensus policy using a natural language verbal description. For example, given Q =*at least* α , if $x > \alpha$ $Q(X) = 1$, this means that a proportion of X fulfils the concept conveyed by the quantifier most, where if $X < \alpha$ $Q(X) = 0$ because the proportion X is not compatible with the concept (the minimum proportion α is not reached) expressed by the quantifier.

However, more generally any function $Q : [0, 1] \to [0, 1]$ such that meets the requirements previously stated for the quantifiers, can be seen to be an appropriate form for generating mediation rules or consensus policies.

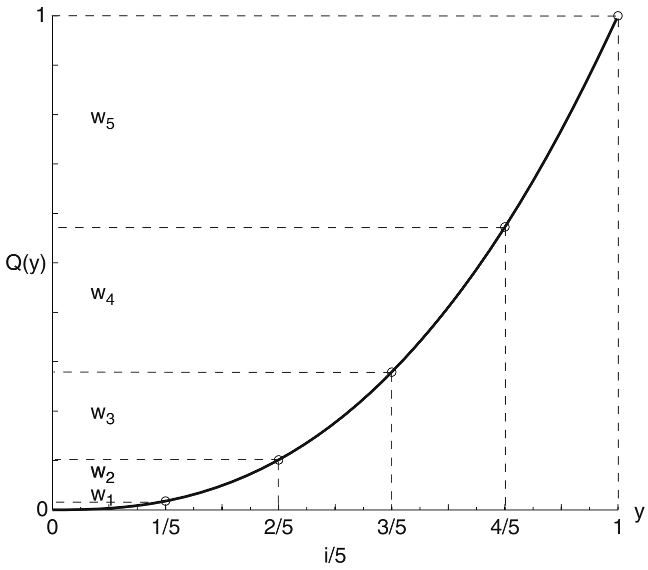

Fig. 2.2 Example of how to obtain the weights from the quantifier for $n = 5$ agents

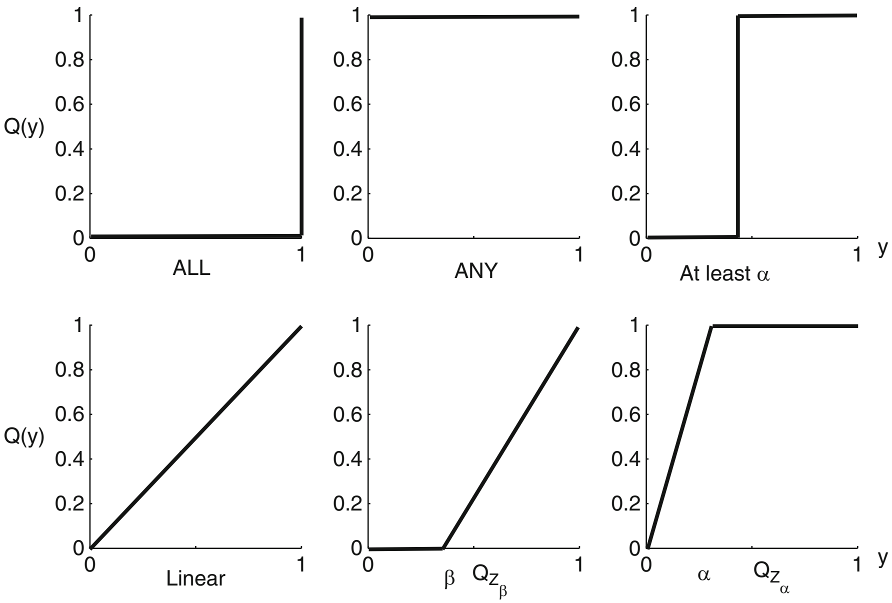

Fig. 2.3 Functional form of typical quantifiers: all, any, at least, linear, piecewise linear Q_{Z_β} and piecewise linear Q_{Z_α}

Table 2.1 VOID values for different quantifiers

Quantifier	VOID
All	1
Any	0
At least α	α
linear	0.5
Q_{Z_α}	$\frac{\alpha}{2}$
Q_{Z_β}	$\frac{1}{2}+\frac{\beta}{2}$
Q_p	$\frac{p}{p+1}$

One feature which distinguishes the different types of mediation rules is the power of an individual agent to eliminate an alternative. For example, in the case of *all* this power is complete, and any agent could force an alternative to be rejected by voting zero. In order to capture this idea, we introduce the *Value Of Individual Disapproval* (VOID) [10], which is defined as:

$$VOID(Q) = 1 - \int_0^1 Q(y)dy \tag{2.3}$$

VOID measures this power of an individual agent to eliminate an alternative. For the *all*, *any, at least* α and *linear* quantifiers the VOID measures are respectively 1, 0, α and 0.5. For the Q_{Z_β} quantifier $VOID(Q_{Z_\beta}) = \frac{1}{2}+\frac{\beta}{2}$ and therefore $VOID(Q_{Z_\beta}) \in [0.5, 1]$. The Q_{Z_α} quantifier gets $VOID(Q_{Z_\alpha}) = \frac{\alpha}{2}$ and $VOID(Q_{Z_\alpha}) \in [0, 0.5]$. Another family of quantifiers are those defined by $Q_p(y) = y^p$ for $p > 0$. In this case $VOID(Q_p) = 1 - \int_0^1 r^p dr = \frac{p}{p+1}$. For Q_p we see that as p increases we get closer to the min and that as p gets closer to zero we get the max (Table 2.1).

2.4.2.3 Computation of the Feedback Contract

Finally, once W has been obtained, the feedback contract at round k is computed as

$$\mathbf{fc}(k) = \mathbf{c_k} + \frac{\mathbf{v}}{\|\mathbf{v}\|} \cdot \sum_{i=1}^{l} w_i \cdot d_{kci}\ , \tag{2.4}$$

where

$$\mathbf{v} = \sum_{i=1}^{l} w_i \cdot \mathbf{r}_{kci}\ . \tag{2.5}$$

Vector $\mathbf{v}$ results from applying the vectorial OWA operator to the direction vectors. The feedback contract is generated in the direction pointed by $\mathbf{v}$ from the origin $\mathbf{c}_k$. The distance at which the feedback contract is generated is obtained by applying the scalar OWA operator to the distances to the centroid. Now, for instance, let us assume a quantifier $Q_p(y) = y^p$ and $p = 20$, which means that

$VOID = 0.95$ (i.e. we want many agents satisfied) and that we have four contracts in the selected cluster. In this case w_l will approach 1 and vector $\mathbf{v}$ will approximate $\mathbf{r}_{kcl}$ pointing to the farther contract from the centroid. However, the feedback contract will be the centroid $\mathbf{c}_k$ because $\sum_{i=1}^{l} w_i \cdot d_{kci} = d_{kcl} = 0$. For a very low $VOID$, w_1 will approximate 1, which means than $\mathbf{v} = \mathbf{r}_{kc1}$ pointing to one of the contracts. In addition, the second summand in $\mathbf{fc}(k)$ will be $\frac{\mathbf{v}}{\|\mathbf{v}\|} \cdot d_{kc1} = min(D_{kc})$, which means that the feedback contract will be very close to one of the contracts (the closer one). These are only two examples of the effect that W has in the generation of the feedback offer. For high $VOID$ values the feedback contract approaches the centroid to satisfy many agents. For low $VOID$ values the feedback contract approaches the closer contracts to the centroid.

2.4.3 Measuring the Quality of the Agreement

Once a feedback contract has been generated, it is important to evaluate how the degree in which this feedback contract satisfies the desired consensus policy. This will serve as an signal to know when to stop the negotiation process. We use the group distance as a measure of closeness to the desired agreement. To compute this group distance, we employ again the OWA weights computed previously and using them we calculate the weighted sum of the distances from the offers in the cluster to the feedback contract. The formula is as follows:

$$Gd_k = \sum_{i=1}^{l} w_i \cdot \|\mathbf{o}_{kci} - \mathbf{fc}(k)\| \ . \tag{2.6}$$

Notice that we use W to OWA-weight the distance estimate to take into account the consensus policy. If the group distance falls below a threshold, the negotiation ends with an agreement on the feedback contract.

2.5 Experimental Evaluation

In this section, we show that the proposed mechanisms provide the mediator the tools to efficiently conduct multiagent negotiations following different consensus policies. In the first experimental setup we have considered seven agents, two issues and two different types of negotiation spaces: a negotiation space where agents' utility functions are strategically built to define a *proof of concept negotiation scenario*, and a *complex negotiation scenario* where utility functions exhibit a more complex structure. In both cases utility functions are built using an aggregation of *Bell functions*. This type of utility functions captures the intuition that agents' utilities for a contract usually decline gradually with distance from their ideal contract. Bell functions are ideally suited to model, for instance, spatial and temporal preferences and to simulate different levels of complexity.

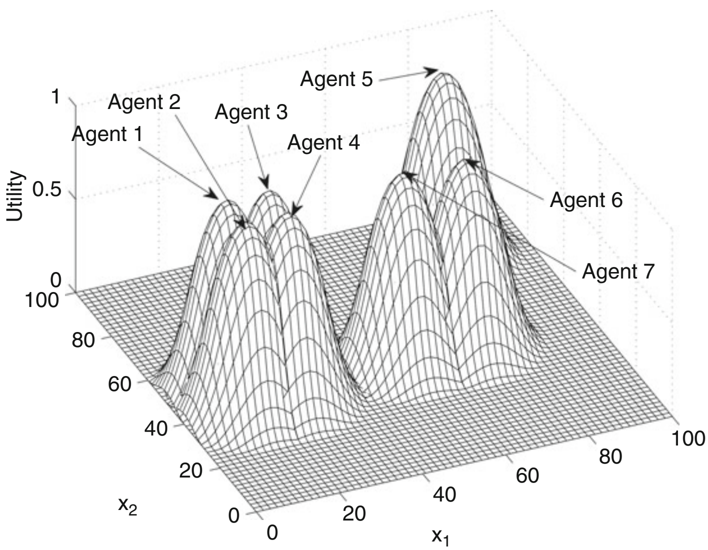

Fig. 2.4 Utility Functions for the proof of concept Scenario

A *Bell* is defined by a center c, height h, and a radius r. Let $\parallel s - c \parallel$ be the Euclidean distance from the center c to a contract s, then the *Bell function* is defined as

$$fbell(s,c,h,r) = \begin{cases} h - 2h\frac{\|s-c\|^2}{r^2} & \text{if } \parallel s-c \parallel < \frac{r}{2} \\ \frac{2h}{r^2}(\parallel s-c \parallel - r)^2 & \text{if } r > \parallel s-c \parallel \geq \frac{r}{2} \\ 0 & \parallel s-c \parallel \geq r \end{cases} \tag{2.7}$$

and the *Bell utility function* as

$$U_{b,s}(s) = \sum_{i}^{nb} fbell(s, c_i, h_i, r_i) \tag{2.8}$$

where nb is the number of generated bells. The complexity of the negotiation space can be modulated by varying c_i, h_i, r_i and nb.

In the *proof of concept negotiation scenario* each agent has a utility function with a single optimum. Figure 2.4 shows in the same graph the agents' utility functions in the bidimensional negotiation space $[0, 100]^2$. Four agents (Agent 1, 2, 3, 4) are in weak opposition (i.e. their preferences are quite similar), Agents 6 and 7 are in weak opposition and in very strong opposition with respect the other agents, and Agent 5 is in very strong opposition with respect the rest of the agents. In the *complex negotiation scenario* (Fig. 2.5) each agent's utility function is generated using two randomly located bells. The radius and height of each bell are randomly distributed

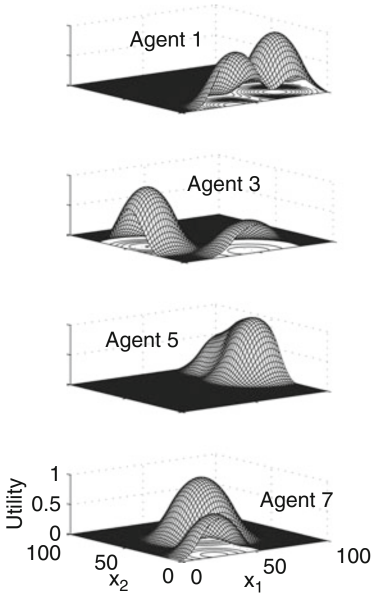

Fig. 2.5 Utility Functions for the Complex Negotiation Scenario

within the ranges $r_i \in [20, 35]$ and $h_i = [0.1, 1]$. The configuration of parameters in the mediator is: $nr = 50$ rounds, $ns = 10$ stages and a group distance threshold 0.001. The cutoffs applied in HC go from 2 in the first stage to 0.1 in the last stage following linear decrements. The probability for an agent to concede (i.e. to attend exclusively the feedback contract) is modelled for each agent using a probability value obtained from a uniform distribution between 0.25 and 0.5. For instance, an agent with probability 0.5 will concede with a 50% probability whenever it is not possible to improve both utility and distance from the feedback contract. We tested the performance of the protocol for three different consensus policies with VOID degrees: 0, 0.5 and 0.95, using the quantifier $Q_p(y) = y^p$. Each experiment consist of 100 negotiations where we capture the utilities achieved by each agent. To analyze the results we first build a 7 agents$\times$100 negotiations utility matrix where each row provides each agent's utilities and each column is a negotiation. The matrix is then reorganized such that each column is individually sorted from higher to lower utility values. Note that after this transformation the association row/particular-agent disappears. Given the matrix, we form seven different utility groups: a first group named *group level 1* where we take the highest utility from each negotiation (i.e. the first row), a second group named *group level 2* with the two first rows and so on. In order to show the performance of the protocol we have used the Kaplan-Meier estimate of the cumulative distribution function (*cdf*) [18] of agents' utilities for each group. Thus, we compute the *cdf* for the highest utilities, for the two highest utilities and so on. The *cdf* estimates the probability of finding agent's utilities below

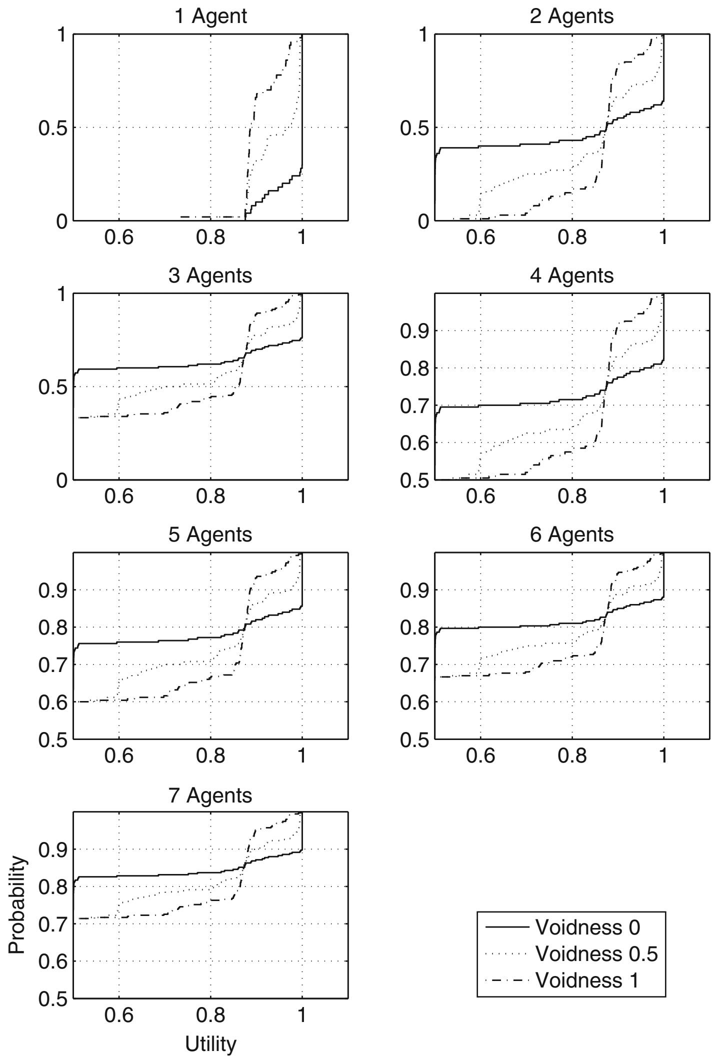

Fig. 2.6 Cumulative distributions of utilities for the *proof of concept scenario*

a certain value. The rationale behind using grouping in the analysis is to evaluate the ability of the protocol to find solutions which satisfy groups of agents.

In the proof of concept scenario (see Fig. 2.4) it can be seen that when an unanimous is needed, the best alternative is to get satisfied agents 1, 2, 3 and 4. If it is enough to have one agent satisfied, any of the utility peaks would be a good solution. In Fig. 2.6 we show the results for the proof of concept scenario. Each line shows the *cdf* for a VOID value, and each plot focuses on the results obtained for each group level. For instance, in group level 1 (i.e. one Agent) there is a 75%

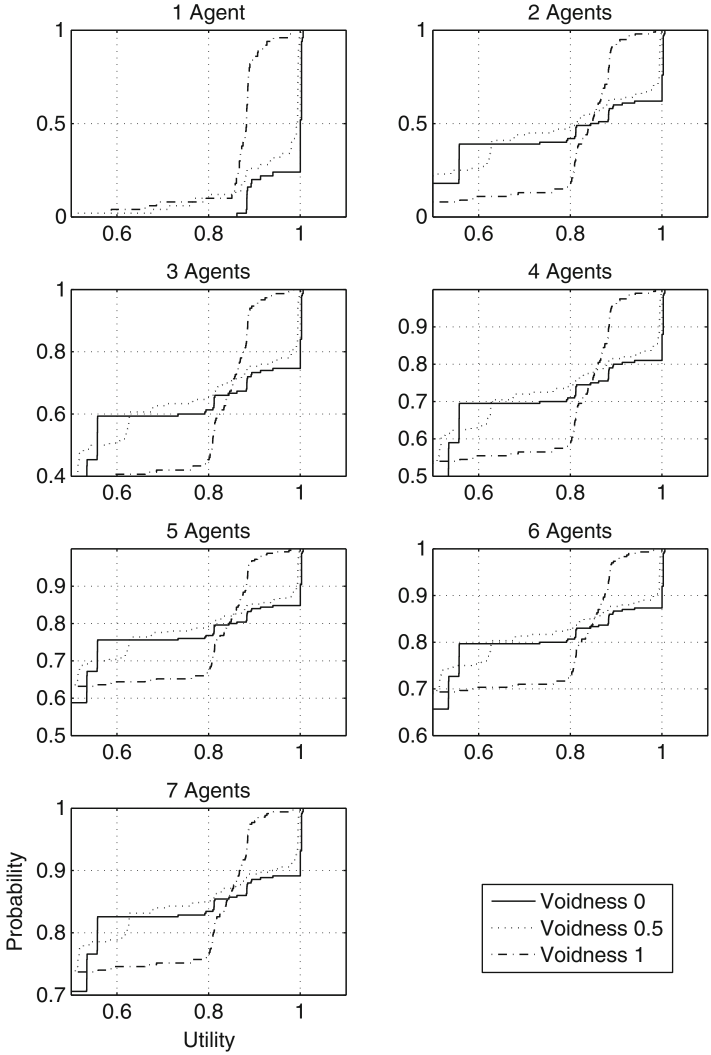

Fig. 2.7 Cumulative distributions of utilities for the *complex negotiation scenario*

probability of having agents with utility 1 for VOID 0, a 40% of having one agent with utility 1 for a VOID 0.5 and a 2% probability of having agents with utility 1 for a VOID approaching 1. We can see how as we evaluate the utility distribution for more agents, if we want many agents satisfied the best we can do is to use a high VOID value. In this case we will share utility in a more uniform way, maybe at the cost of not having agents highly satisfied.

In Fig. 2.7 the results for the complex negotiation scenario are shown. The results also show that as VOID increases, the mediator biases the search for agreements where more agents are satisfied at the expense of the individual satisfaction level.

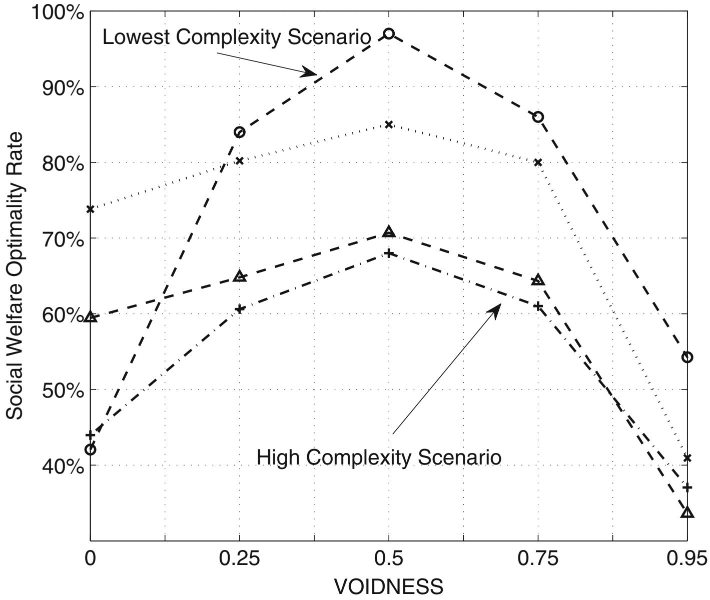

Fig. 2.8 Social Welfare Optimality Rate vs VOID

In general, it is worth noting that the application of a consensus policy may incur in a cost in terms of social welfare. In a second experimental setup we have considered seven agents, two issues and four different types of negotiation spaces in increasing complexity to evaluate this issue.

Figure 2.8 shows the social welfare measurements (sum of utilities) for different VOID degrees. Social welfare is normalized to its optimal value. VOID ranges from 0 to 0.95. We can see how the application of consensus policies come at a cost in terms of social welfare, both for low and for high VOID values. For example, in scenarios where there exist a strong opposition among the agents, if we want to have many agents satisfied, individual utilities cannot be simultaneously large for all the agents, and therefore social welfare decreases. Also note that there exists a VOID value which maximizes social welfare. For complex scenarios, there will be a trade-off between VOID and social welfare.

2.6 Conclusion

We argue that there exist situations where an unanimous agreement is not possible or simply the rules imposed by the system may not seek such unanimous agreement. Thus, we developed a hierarchical consensus policy based mediation framework (HCPMF) to perform multiparty negotiations. To perform the exploration of the negotiation space agents use a variation of the GPS non-linear optimization technique. The mediator guides the joint exploration of a solution by using aggregation rules which take the form of linguistic expressions. These rules are applied over the agents' offered contracts in order to generate a feedback contract which is submitted to the agents in order to guide their exploration. To avoid zones of no agreement the mediator uses Hierarchical Clustering to form clusters of agents. We showed empirically that HCPMF efficiently manages negotiations following predefined consensus policies, which has been modelled using OWA operators.

The negotiation framework presented is one of the first proposals that incorporate alternate consensus definitions for the mediation rule as an integral part of multiparty negotiation protocols. This framework can be extended to incorporate more complex consensus rules that would take into consideration, for instance, the different importance of the negotiating agents or their attitudes. There are also open aspects that we expect to deal with in future works. It is expected that the performance of the protocol deviates from the optimal if agents act strategically. Alternatives ways of generating the feedback contract, based for instance on the history of passed offers, and not only on their current position should be considered. Finally, we plan to explore its possible application to domains as consortium formation in brokering events.

Acknowledgements This work has been supported by Spanish Ministry of Economy and Innovation grant IPT-2012-0808-370000, STIMULO research project.

References

1. Hindriks, K., Jonker, C., Tykhonov, D.: A multi-agent environment for negotiation. In: El Fallah Seghrouchni, A., Dix, J., Dastani, M., Bordini, R.H. (eds.) Multi-Agent Programming, pp. 333–363. Springer, New York (2009)
2. Klein, M., Faratin, P., Sayama, H., Bar-Yam, Y.: Protocols for negotiating complex contracts. IEEE Intell. Syst. **18**(6), 32–38 (2003)
3. Endriss, U., Maudet, N., Sadri, F., Toni, F.: Negotiating socially optimal allocations of resources. J. Artif. Intell. Res. **25**, 315–348 (2006)
4. Lai, G., Sycara, K.: A generic framework for automated multi-attribute negotiation. Group Decis. Negot. **18**, 169–187 (2009)
5. Ehtamo, H., Hamalainen, R.P., Heiskanen, P., Teich, J., Verkama, M., Zionts, S.: Generating pareto solutions in a two-party setting: constraint proposal methods. Manag. Sci. **45**(12), 1697–1709 (1999)
6. Heiskanen, P., Ehtamo, H., Hamalainen, R.P.: Constraint proposal method for computing pareto solutions in multi-party negotiations. Eur. J. Oper. Res. **133**(1), 44–61 (2001)
7. Li, M., Vo, Q.B., Kowalczyk, R.: Searching for fair joint gains in agent-based negotiation. In: Decker, K., Sichman, J., Sierra, C., Castelfranchi, C. (eds.) Proceedings of 8th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2009), Budapest, pp. 1049–1056, 10–15 May 2009
8. Kacprzyk, J.: Group decision making with a fuzzy linguistic majority. Fuzzy Sets Syst. **18**(2), 105–118 (1986)
9. Lewis, R.M., Torczon, V., Trosset, M.W.: Direct search methods: then and now. J. Comput. Appl. Math. **124**, 191–207 (2000)

10. Yager, R., Kacprzyk, J.: The Ordered Weighted Averaging Operators: Theory and Applications. Kluwer, Dordrecht (1997)
11. Yager, R.: Quantifier guided aggregation using OWA operators. Int. J. Intell. Syst. **11**, 49–73 (1996)
12. Ward, J.H.: Hierarchical grouping to optimize an objective function. J. Am. Stat. Assoc. **58**(301), 236–244 (1963)
13. Yager, R.: E-z OWA weights. In: Proceedings of 10th International Fuzzy Systems Association World Congress, Istanbul, pp. 39–42 (2003)
14. Grabisch, M., Orlovski, S.A., Yager, R.R.: Fuzzy Sets in Decision Analysis, Operations Research and Statistics, pp. 31–68. Kluwer, Norwell (1998)
15. Yager, R.R., Filev, D.P.: Essentials of Fuzzy Modeling and Control. Wiley-Interscience, New York (1994)
16. Zadeh, L.: A computational approach to fuzzy quantifiers in natural languages. Comput. Math. Appl. **9**, 149–184 (1983)
17. Liu, X., Han, S.: Orness and parameterized RIM quantifier aggregation with OWA operators: a summary. Int. J. Approx. Reason. **48**(1), 77–97 (2008)
18. Kaplan, E.L., Meier, P.: Nonparametric estimation from incomplete observations. J. Am. Stat. Assoc. **53**(282), 457–481 (1958)

http://www.springer.com/978-4-431-54757-0

Novel Insights in Agent-based Complex Automated Negotiation

Marsa-Maestre, I.; Lopez-Carmona, M.A.; Ito, T.; Zhang, M.; Bai, Q.; Fujita, K. (Eds.)

2014, X, 204 p. 50 illus., 15 illus. in color., Hardcover
ISBN: 978-4-431-54757-0