Negotiation Tactics for Autonomous Agents

Fernando Lopes*, Nuno Mamede**, A. Q. Novais*, Helder Coelho***

*INETI, Est. Paco Lumiar, 1699 Lisboa Codex, Portugal
  {flopes, anovais@dms.ineti.pt}
**IST, Av. Rovisco Pais, 1049-001 Lisboa, Portugal
  Nuno.Mamede@acm.org
***Fac. Ciências, Campo Grande, 1700 Lisboa, Portugal
  hcoelho@di.fc.ul.pt

Abstract

Autonomous agents are being increasingly used in a wide range of applications. The agents operate in common environments and, over time, conflicts inevitably occur among them. Negotiation is the predominant process for solving conflicts. Recent growing interest in electronic commerce has also given increased importance to negotiation. This paper presents a generic negotiation mechanism that handles multi-party, multi-issue and single or repeated rounds and introduces a set of negotiation tactics that express the initial attitude of the agents and generate counterproposals either by making or not making concessions.

1. Introduction

Autonomous agents are being increasingly used in a wide range of applications. The agents have a high degree of control over their internal state and behavior - they can decide for themselves which goals to adopt, which actions to perform in order to achieve these goals, and when to perform these actions.

Most applications involve or require multiple agents operating in complex environments and, over time, conflicts inevitably occur among them. Most conflicts arise because the agents adopt goals that are per se incompatible or there are limitations in the resources needed to fulfill the adopted goals.

The predominant process for solving conflicts is negotiation - the process by which the parties attempt to influence one another to achieve their needs, while at the same time taking the needs of the others into account [4].

This paper makes two main contributions towards the goal of developing autonomous agents with competence for solving conflicts via negotiation. The first is to present a generic negotiation mechanism that handles multi-party, multi-issue and single or repeated rounds. The mechanism supports problem restructuring allowing the dynamic addition of negotiation issues. The second contribution is to introduce a set of negotiation tactics that express the initial attitude of the agents and generate counterproposals either by making or not making concessions.

This paper builds on our previous work [5, 6, 7]. In these papers, we described methods for detecting and validating conflicts, presented the main prenegotiation activities, introduced the negotiation mechanism, and defined concession tactics. In this paper, we continue the description of the negotiation mechanism and introduce a set of negotiation tactics.

The remainder of the paper is structured as follows. Section 2 presents a generic model of autonomous agents. The model defines the individual behavior of agents and forms a basis for the development of negotiating agents. Section 3 addresses the operational and strategic process of planning and preparing for negotiation. Section 4 describes a generic negotiation mechanism. Section 5 introduces a set of negotiation tactics. Section 6 situates this work within the related literature. Section 7 concludes and outlines future avenues of research.

2. Autonomous Agents

Let $\text{Agents} = \{a_1, a_2, \ldots, a_n\}$ be a set of autonomous agents. This section presents a brief description of the key features of every agent $a_i \in \text{Agents}$ (see our earlier work for an in-depth discussion [5, 6]).

The agent $a_i$ has a set $B_i = \{b_{i1}, b_{i2}, \ldots\}$ of beliefs and a set $G_i = \{g_{i1}, g_{i2}, \ldots\}$ of goals. Beliefs represent...
information about the world and the agent himself. Goals represent world states to be achieved.

The agent \( ag_i \) has a library \( PL_i=\{p_{11},p_{12},\ldots,\} \) of plan templates representing simple procedures for achieving goals. The library has composite plan templates specifying the decomposition of goals into more detailed subgoals, and primitive plan templates specifying actions directly executable by \( ag_i \). Every plan template \( pt_{ik}\) has a header: \( header_{i(k)}=<pname_{ikl}, pvars_{ikl}\)>, where \( pname_{ikl} \) is the name of \( pt_{ikl} \) and \( pvars_{ikl} \) is a set of variables (parameters of \( pt_{ikl} \)), and also a number of other components [6].

The agent \( ag_i \) is able to generate complex plans from the simpler plan templates stored in the library. A plan \( p_{ik} \) for achieving a goal \( g_{ik}\) is a 3-tuple: \( p_{ik}=<PT_{ik}\), \( \leq h \), \( \leq \rangle \), where \( PT_{ik} \) is a list of plan templates, \( \leq h \) is a binary relation establishing a hierarchy on \( PT_{ik} \), and \( \leq \rangle \) is another binary relation establishing a temporal order on \( PT_{ik} \). The plan \( p_{ik} \) is represented as a hierarchical and temporally constrained And-tree denoted by \( Pstruct_{ik} \).

At any instant, the agent \( ag_i \) has a number of plans for execution. These plans are the plans currently adopted by \( ag_i \) and are stored in the intention structure \( IS_i=\{p_{11},p_{12},\ldots,\} \). As stated above, a plan \( p_{ik}\) is a 3-tuple: \( p_{ik}=<PT_{ik}, \leq h, \leq \rangle \). For each plan template \( pt_{ikl}\), the header of \( pt_{ikl} \) is referred as \( intention_{ikl} \) formulated by \( ag_i \).

The agent \( ag_i \) has information about the agents in Agents. The information is stored in the social description \( SD_i=\{SD_{ag1},\ldots,SD_{agn}\} \). Each entry \( SD_{ag}(ag_i)=<B_{ag}(ag_i).G_{ag}(ag_i).I_{ag}(ag_i)> \), \( SD_{ag}(ag_i) \in SD_i \), contains the beliefs, goals and intentions that \( ag_j \) believes \( ag_i \) has.

3. Planning and Preparing for Negotiation

Autonomous agents often operate in common environments and, over time, conflicts inevitably occur among them. Let \( Ag=\{ag_1,\ldots,ag_n\}, Ag\subset\text{Agents} \), be a set of autonomous agents. Let \( P_{Ag}=[P_{11},P_{12},\ldots,P_{nm}] \) be a set of plans of the agents in \( Ag \) including intentions \( I_{Ag}=[int_{11},\ldots,int_{ik},\ldots,int_{nm}] \), respectively. Let the intentions in \( I_{Ag} \) represent commitments to achieve exclusive world states. In this situation, there is a conflict among the agents in \( Ag \).

Negotiation is the predominant process for resolving conflicts. Successful negotiators agree on one thing: the key to success in negotiation is planning and preparation ([14]). This section presents a brief description of the main activities that each agent \( ag_i \) must attend to in order to plan and prepare for negotiation (see our earlier work for an in-depth discussion [7]).

3.1. Negotiation Problem Structure Generation

Conflicts raise negotiation problems. Let \( B_i \) and \( G_i \) be the sets of beliefs and goals of \( ag_i \), respectively. Let \( p_{ik}\in P_{agi} \) be a plan of \( ag_i \) for achieving goal \( g_{ik}\) in \( G_i \). Let \( int_{ikm}\in GI_{agi} \) be an intention of \( p_{ik} \). Let \( A=Ag-\{ag_i\} \) and \( \lambda_i=I_{Ag}\{int_{ikm}\} \). A negotiation problem from the perspective of \( ag_i \) is a 6-tuple: \( NP_{ik}=<ag_i, B_i, G_i, int_{ikm}, A, \lambda_i> \).

The problem \( NP_{ik} \) has a structure \( NStruct_{ik} \) consisting of a hierarchical And-Or tree ([7]). The nodes of the tree are plan templates. The header of the root node describes the goal \( g_{ik} \) (called negotiation goal). Formally, \( NStruct_{ik} \) is a 4-tuple: \( NStruct_{ik}=<NPT_{ik}, \leq h, \leq \rangle, \leq_{a} \rangle, NPT_{ik} \) is a list of plan templates, \( \leq h \) and \( \leq \rangle \) have the meaning just specified, and \( \leq_{a} \rangle \) is a binary relation establishing alternatives among the plan templates in \( NPT_{ik} \). The structure \( NStruct_{ik} \) defines all the possible solutions of \( NP_{ik} \) currently known by \( ag_i \). A possible solution is a plan that can achieve \( g_{ik} \).

3.2. Issue Identification and Prioritization

The negotiation issues of \( ag_i \) are obtained from the leaves of \( NStruct_{ik} \). Let \( LI_{ik}=\{pt_{ik1},pt_{ik2},\ldots,\} \) be the collection of plan templates constituting the leaves of \( NStruct_{ik} \). The header \( (pname_{ikl} \) and \( pvars_{ikl} \)) of every plan template \( pt_{ikl}\in LI_{ik} \) is called a fact and denoted by \( f_{ikl} \). Formally, a fact \( f_{ikl} \) is a 3-tuple: \( f_{ikl}=\{is_{ikl},v_{is_{ikl}},r_{ikl}\} \), where \( is_{ikl} \) is a negotiation issue (corresponding to \( pname_{ikl} \)), \( v_{is_{ikl}} \) is a value of \( is_{ikl} \) (corresponding to an element of \( pvars_{ikl} \)), and \( r_{ikl} \) is a list of arguments (corresponding to the remaining elements of \( pvars_{ikl} \)). Typically, \( r_{ikl} \) is an empty list (e.g., \{<price,50>\}).

Let \( F_{ik}=[f_{ik1},\ldots,f_{ikn}] \) be the set of facts of \( NStruct_{ik} \). The negotiating agenda of \( ag_i \) is the set of issues
\[ I_{ik} = \{i_{k1}, \ldots, i_{kz}\} \] associated with the facts in \( F_{ik} \) (for clarity, we consider that every fact in \( F_{ik} \) defines a different issue). The interval of legal values for each issue \( i_{kj} \in I_{ik} \) is represented by \( D_{ikj} = [\min_{ikj}, \max_{ikj}] \).

For each issue \( i_{kj} \in I_{ik} \), let \( w_{ikj} \) be a real number called \textit{importance weight} that represents its relative importance. Let \( W_k = \{w_{ik1}, \ldots, w_{ikz}\} \) be the set of importance weights of the issues in \( I_{ik} \). The importance weights are normalized, \( \sum_{j=1}^{z} w_{ikj} = 1 \). The \textit{priority} of the issues in \( I_{ik} \) is just defined as their relative importance.

### 3.3. Limits and Aspirations Formulation

A limit or reservation value is a bargainer’s ultimate fallback position, the level of benefit beyond which he is unwilling to concede. Aspiration is the benefit sought at any particular time. Limit tends to remain constant over time, whereas aspiration declines toward limit [8].

Limits and aspirations are formulated for each issue at stake in negotiation. The limit for issue \( i_{kj} \in I_{ik} \) is represented by \( \lim_{ikj} \) and the initial aspiration by \( \text{asp}_{ikj} \), with \( \lim_{ikj} \leq \text{asp}_{ikj} \leq D_{ikj} \). Let \( \text{asp}, \lim \in [0, 1] \) be the level of benefit beyond which he is unwilling to concede.

### 3.4. Negotiation Constraints Definition

Negotiation constraints bound the acceptable values for the issues in \( I_{ik} \). \textit{Hard constraints} are linear boundary constraints that specify threshold values for the issues. They cannot be relaxed. \textit{Soft constraints} are linear boundary constraints that specify minimum acceptable values for the issues. They can be relaxed, if necessary. They also can have different degrees of flexibility.

Constraints are defined for each issue \( i_{kj} \in I_{ik} \). The hard constraint \( h_{ckj} \) for \( i_{kj} \) has the form:

\[
\begin{align*}
    h_{ckj}(i_{kj}) &= \left( i_{kj} \geq \lim_{ikj}, \text{flex}=0 \right),
    \text{where flex}=0 \text{ represents null flexibility (inflexibility).}
\end{align*}
\]

The soft constraint \( sc_{ckj} \) for \( i_{kj} \) has the similar form:

\[
\begin{align*}
    sc_{ckj}(i_{kj}) &= \left( i_{kj} \geq \text{asp}_{ikj}, \text{flex}=n \right),
    \text{where flex}=n, n \in N, \text{represents the degree of flexibility of sc}_{ckj}.
\end{align*}
\]

### 3.5. Negotiation Strategy Selection

The agent \( a_{gi} \) has a library \( \text{SL}_i = \{\text{str}_{i1}, \text{str}_{i2}, \ldots\} \) of negotiation strategies and a library \( \text{TL}_i = \{\text{tact}_i1, \text{tact}_i2, \ldots\} \) of negotiation tactics. \textit{Negotiation strategies} are functions that define the negotiation tactics to be used throughout the negotiation process (e.g., starting high and conceding slowly). \textit{Negotiation tactics} are functions that define the actions or moves to be made at each point of the negotiation process (see section 5).

Strategy selection is an important task and must be carefully planned [4, 8, 9]. The strategy most suitable for a particular negotiation situation often depends on the situation itself and cannot be specified in advance. As a result, strategy selection is a difficult task. In this paper, we just assume that \( a_{gi} \) selects a strategy \( \text{str}_{ik} \in \text{SL}_i \) that he considers appropriate accordingly to his experience.

### 4. The Negotiation Mechanism

This section presents a domain-independent description of a generic negotiation mechanism.

#### 4.1. Overview

The mechanism is shown in Fig. 1 from the perspective of an agent \( a_{gi} \in A_g \) that communicates a negotiation proposal. Let \( \text{NP}_{ik} \) represent \( a_{gi} \) ’s perspective of a negotiation problem and \( \text{NPstruct}_{ik} \) be the structure of \( \text{NP}_{ik} \).

First, \( a_{gi} \) generates the initial negotiation proposal set \( \text{INPS}_{ik} = \{\text{prop}_{ik1}, \text{prop}_{ik2}, \ldots\} \), i.e., the set of negotiation proposals satisfying the requirements imposed by \( \text{NPstruct}_{ik} \). Broadly speaking, a negotiation proposal \( \text{prop}_{ikm} \in \text{INPS}_{ik} \) is a set of facts (see subsection 4.2). Next, \( a_{gi} \) determines the initial acceptable proposal set \( \text{IAPS}_{ik} \). \( \text{IAPS}_{ik} \subseteq \text{INPS}_{ik} \), i.e., the set of acceptable proposals. An acceptable proposal is a negotiation proposal that satisfies both the hard and soft negotiation constraints (see subsection 4.3). Next, \( a_{gi} \) evaluates the acceptable proposals in \( \text{IAPS}_{ik} \) using an additive scoring function, and selects a particular proposal \( \text{prop}_{ikm} \) accordingly to his negotiation strategy \( \text{str}_{ik} \) (see subsection 4.4).

Following this, \( a_{gi} \) communicates the proposal \( \text{prop}_{ikm} \) to all the agents in \( A \). Each agent \( a_{gj} \in A \) then evaluates \( \text{prop}_{ikm} \) and either: (i) accepts \( \text{prop}_{ikm} \); (ii) breaks off negotiation, (iii) rejects \( \text{prop}_{ikm} \) without making a critique, or (iv) rejects \( \text{prop}_{ikm} \) and makes a critique. Broadly speaking, a critique is a statement of aspirations, priorities of the issues, etc. The tasks performed by each agent \( a_{gj} \) are not shown in Fig. 1.
Next, $ag_i$ processes the responses and checks whether a negotiation agreement was reached. Generally speaking, a negotiation agreement is a proposal accepted by all the agents. So, if the proposal $prop_{ikm}$ is accepted by all agents in $A$, the negotiation ends. Otherwise, $ag_i$ checks whether any of the agents in $A$ decided to break off negotiation. If at least one agent broke off, the negotiation ends. If not, $ag_i$ determines whether or not to break off negotiation unilaterally. If so, the negotiation ends. Otherwise, $ag_i$ checks whether the negotiation deadline was reached. Again, if so, the negotiation ends.

If the deadline was not reached, $ag_i$ may decide either:
(i) to do nothing (inaction), or (ii) to prepare a new proposal $prop_{ikm+1}$. The preparation of $prop_{ikm+1}$ can be done either: (i) by modifying the rejected proposal $prop_{ikm}$ (see section 5), or (ii) by selecting a new proposal from $IAPS_{ik}$. The negotiation strategy $str_{ik}$ of $ag_j$ defines the particular method to use. The new proposal $prop_{ikm+1}$ is then communicated to all agents in $A$ and the tasks just described are repeated.

The decision to do nothing closes one round of negotiation. Negotiation proceeds to a new round in which another agent $ag_j \in A$ generates and communicates a counterproposal. Broadly speaking, a counterproposal is a proposal made in response to a previous proposal. This is then repeated for all the agents in $A$.

It is worth pointing out that each agent $ag_i$ in $Ag$ may decide either: (i) to relax the soft constraints, or (ii) to restructure the negotiation problem. This can be done at each point of the negotiation process (fig. 1 shows only constraint relaxation and problem restructuring at the beginning of negotiation). Problem restructuring allows the dynamic addition and remotion of negotiation issues.

4.2. Negotiation Proposal Generation

Negotiation proposal generation is a process that takes $NP_{\text{struct}_{ik}}$ as input and generates the set $INPS_{ik}$ of negotiation proposals through an iterative procedure involving three main tasks: (i) problem interpretation, (ii) proposal preparation, and (iii) proposal addition.

Let $g_{ik}$ be the negotiation goal of $ag_i$. Let $F_{ik} = \{f_{ik1}, \ldots, f_{ikz}\}$ be the set of facts of $NP_{\text{struct}_{ik}}$ and $I_{ik} = \{is_{ik1}, \ldots, is_{ikz}\}$ be the set of issues associated with the facts in $F_{ik}$.

Problem interpretation consists of searching $NP_{\text{struct}_{ik}}$ for any possible solution $p_{ik}$ of $NP_{ik}$ and selecting the primitive plan templates of $p_{ik}$. More specifically, the search starts at the root node of
Proposal preparation consists of determining a negotiation proposal proposal\_ikm = \{i_{ik}^{1},\ldots,i_{ik}^{f}\}, \text{prop}_{ikm} \subseteq F_{ik}, i.e., a set of facts corresponding to the headers of the primitive plan templates in \text{pp}_{ik}. This task is formalized by a function \text{prepare\_proposal} which takes the set \text{pp}_{ik} as input and returns \text{prop}_{ikm}.

The preparation of a proposal \text{prop}_{ikm} partitions the set \mathcal{F}_{ik} of facts into: (i) subset \text{prop}_{ikm}, and (ii) subset \text{pcompl}_{ikm} = \{f_{ik}^{1},\ldots,f_{ik}^{f}\}, called proposal complement of \text{prop}_{ikm}, corresponding to the remaining facts of \mathcal{F}_{ik}.

The facts in \text{prop}_{ikm} are fundamental for achieving the negotiation goal \text{g}_{ik}. They are the inflexible facts of negotiation, for proposal \text{prop}_{ikm}. The negotiation issues \text{Iprop}_{ikm} = \{i_{ik}^{1},\ldots,i_{ik}^{f}\} associated with these facts are the inflexible issues. On the other hand, the facts in \text{pcompl}_{ikm} are not important for achieving \text{g}_{ik}. They are the flexible issues of negotiation, for proposal \text{prop}_{ikm}.

The issues \text{Icompl}_{ikm} = \{i_{ik}^{1},\ldots,i_{ik}^{f}\} associated with these facts are the flexible or bargaining issues.

Proposal addition consists of adding the negotiation proposal \text{prop}_{ikm} to the set \text{INPS}_{ik}. This task is formalized by a function \text{add\_proposal} which takes \text{INPS}_{ik} and \text{prop}_{ikm} as input and returns \text{INPS}_{ik} + \text{prop}_{ikm}.

4.3. Acceptable Proposal Preparation

Acceptable proposal preparation involves two main tasks: (i) feasible proposal formulation, and (ii) acceptable proposal determination. Let \text{prop}_{ikm} = \{f_{ik}^{1},\ldots,f_{ik}^{f}\} be a negotiation proposal. Let \text{Iprop}_{ikm} = \{i_{ik}^{1},\ldots,i_{ik}^{f}\} be the set of issues associated with facts in \text{prop}_{ikm}. Let \text{HCprop}_{ikm} = \{h_{ik}^{1},\ldots,h_{ik}^{f}\} and \text{SCprop}_{ikm} = \{s_{ik}^{1},\ldots,s_{ik}^{f}\} be the sets of hard and soft constraints for issues in \text{Iprop}_{ikm}, respectively. Let \text{Iprop}_{ikm} = \{lim_{ik}^{1},\ldots,lim_{ik}^{f}\} be the limits for issues in \text{Iprop}_{ikm}. Let \text{Aprop}_{ikm} = \{asp_{ik}^{1},\ldots,asp_{ik}^{f}\} be the initial aspirations of \text{ag}_{i} for issues in \text{Iprop}_{ikm}.

Feasible proposal formulation consists of generating the set \text{IFPS}_{ik}, \text{IFPS}_{ik} \subseteq \text{INPS}_{ik} of feasible proposals. A negotiation proposal \text{prop}_{ikm} \in \text{INPS}_{ik} is feasible if the issues in \text{Iprop}_{ikm} satisfy the set \text{HCprop}_{ikm} of hard constraints. This task is formalized by a function \text{feasible\_proposals} which takes \text{INPS}_{ik} as input and returns \text{IFPS}_{ik}.

Acceptable proposal determination consists of generating the set \text{IAPS}_{ik}, \text{IAPS}_{ik} \subseteq \text{IFPS}_{ik}, of acceptable proposals. A feasible proposal \text{prop}_{ikm} is acceptable if the issues in \text{Iprop}_{ikm} satisfy the set \text{SCprop}_{ikm} of soft constraints. This task is formalized by a function \text{acceptable\_proposals} which takes \text{IFPS}_{ik} as input and returns \text{IAPS}_{ik}.

4.4. Proposal Evaluation and Selection

Proposal evaluation consists of computing a score for each proposal in \text{IAPS}_{ik} and ordering the proposals in a descending order of preference. This task is formalized by a function \text{evaluate\_proposals} which takes \text{IAPS}_{ik} as input, computes a score \text{V}_{\text{prop}_{ikm}} \in \mathbb{R} for each acceptable proposal \text{prop}_{ikm} \in \text{IAPS}_{ik} reflecting the preference of \text{ag}_{i}, and returns the ordered set \text{IAPS}_{ik}.

The score of each proposal \text{prop}_{ikm} is computed using an additive scoring function \cite{9}. Let \text{W}_{ik} = \{w_{ik}^{1},\ldots,w_{ik}^{f}\} be the set of importance weights of the issues in \text{Iprop}_{ikm}. Let \text{C}_{ikm} = \{v(i_{ik}^{1}),\ldots,v(i_{ik}^{f})\} be the values of the issues in \text{Iprop}_{ikm} (\text{C}_{ikm} is called a contract). For each issue \text{i}_{ik} \in \text{Iprop}_{ikm} defined over the interval \text{D}_{ik} = [\text{min}_{ik},\text{max}_{ik}]$, let \text{V}_{ikl} be a component scoring function that gives the score that \text{ag}_{i} assigns to a value \text{v}_{ik} \in \text{D}_{ik} of \text{i}_{ik}. The score for contract \text{C}_{ikm} is given by a function \text{V}:

\[
\text{V}(\text{C}_{ikm}) = \sum_{j=1}^{p} w_{ikj} \text{V}_{ikj}(v(i_{ikj}))
\]

The proposal \text{prop}_{ikm} is identified with contract \text{C}_{ikm} and both have the same score.

Proposal selection consists of selecting a particular proposal \text{prop}_{ikm} \in \text{IAPS}_{ik}. This task is formalized by a function \text{select\_proposal} which takes \text{IAPS}_{ik}, the negotiation strategy \text{str}_{ik} and the library of tactics \text{TL}_{ik} as input and returns a proposal \text{prop}_{ikm}. The negotiation
strategy $str_i$ dictates a specific tactic $tact_i \in TL_i$ to use. The tactic $tact_i$ specifies a particular proposal $prop_{ikm}$.

4.5. Proposal Modification

Proposal modification consists of computing a new proposal $prop_{ikm+1}$ from a rejected proposal $prop_{ikm}$. This task is formalized by a function $modify\_proposal$ which takes $prop_{ikm}$, $str_i$ and $TL_i$ as input and returns a new proposal $prop_{ikm+1}$. Again, the strategy $str_i$ defines a specific tactic $tact_i \in TL_i$ to use. The tactic $tact_i$ modifies $prop_{ikm}$ to make it more acceptable.

5. Negotiation Tactics

Negotiation tactics are functions that define the actions or moves to be made at each point of the negotiation process. This section describes a set of tactics from the perspective of each agent $ag_i \in Ag$.

5.1. Opening Negotiation Tactics

Opening negotiation tactics are functions that express the initial attitude of $ag_i$ and specify the proposal to submit at the beginning of negotiation. In this paper, we consider the following three tactics:

1. starting high – expresses an aggressive opening attitude and specifies the proposal with the highest score;
2. starting optimistic – expresses an optimistic opening attitude and specifies a proposal with a score between the highest and the lowest;
3. starting realistic – expresses a realistic opening attitude and specifies the proposal with the lowest score.

Let $IAPS_{ik} = \{prop_{ik1}, prop_{ik2}, \ldots\}$ be the set of acceptable proposals of $ag_i$ ordered in a descending order of preference ($prop_{ik1}$ is the proposal with the highest score $Vprop_{ik}$). The tactic starting high is formalized by a function $starting\_high$ which takes $IAPS_{ik}$ as input and returns $prop_{ik1}$, i.e.,

$$starting\_high(IAPS_{ik}) = prop_{ik1} \mid \forall prop_{ikj} \in IAPS_{ik}, Vprop_{ikj} \geq Vprop_{ik1}$$

The definition of the functions for the tactics starting optimistic and starting realistic is essentially identical to that of starting high and is omitted.

5.2. Bargaining Issue Manipulation Tactic

Successful negotiators often add to the negotiation agenda issues that they do not really care about in the hope that the other parties will feel strongly about these issues - strong enough to be willing to make compensating concessions [9].

Bargaining issue manipulation is a tactic that allows $ag_i$ to act strategically by using the flexible facts (bargaining issue with specific values) to extract concessions from the other parties. Let $prop_{ikm} = [f_{ik1}, \ldots, f_{ikp}]$ be a negotiation proposal submitted by $ag_i$ to the agents in $Ag$ and rejected. Let $pcompl_{ikm} = [f_{ikp+1}, \ldots, f_{ikz}]$ be the complement of $prop_{ikm}$.

Bargaining issue manipulation allows $ag_i$ to add to the negotiation agenda an issue that aims to extract concessions from the other parties. Let $v_1[is_{ikj}]=prop_{ikm+1}$ and $v_2[is_{ikj}]=prop_{ikm} \cdot f_{ikj}$.

where $\cdot$ stands for concatenation.

Bargaining issue manipulation is a non-concession tactic. Indeed, the rejected proposal $prop_{ikm}$ and the new proposal $prop_{ikm+1}$ have very similar scores (hence, $ag_i$ does not make a concession).

5.3. Concession Tactics

Concession tactics are functions that compute new values for each negotiation issue. The tactics model the concessions to be made on every issue at each point of the negotiation process.

Let $I_{ik}$ be the set of negotiation issues. A concession on an issue $is_{ikj} \in I_{ik}$ is a change in the value of $is_{ikj}$ that reduces the level of benefit sought. In this paper, we consider the following five tactics:

1. stalemate - models a null concession on $is_{ikj}$;
2. tough - models a small concession on $is_{ikj}$;
3. moderate - models a moderate concession on $is_{ikj}$;
4. soft - models a large concession on $is_{ikj}$;
5. compromise - models a complete concession on $is_{ikj}$.

Let $prop_{ikm}$ be a proposal submitted by $ag_i$ and rejected. Let $v[is_{ikj}]=prop_{ikm+1}$ be the value of $is_{ikj}$ offered in $prop_{ikm}$. Let $lim_{ikl}$ be the limit for $is_{ikj}$. Let $V_{ikj}$ be the component
scoring function of $a_{gi}$ for is$_{ijk}$. Let $v(is_{ijk})_{\text{new}}$ be the new value of is$_{ijk}$ to be offered in a new proposal prop$_{i,m+1}$. The five tactics are formalized by the following expression:

$$v(is_{ijk})_{\text{new}} = v(is_{ijk})_{\text{old}} + (-1)^{w} F \left| \lim_{ijk} - v(is_{ijk})_{\text{old}} \right|$$

where $w=0$ if $V_{ijk}$ is monotonically decreasing or $w=1$ if $V_{ijk}$ is monotonically increasing, and $F \in [0, 1]$ is a factor.

The factor $F$ can be simply a constant [2]. The five tactics are then defined by considering different values for $F$. For instance, the stalemate tactic is defined by setting $F=0$, the tough tactic by $F \in [0, 0.5]$, the moderate tactic by setting $F=0.5$, the soft tactic by $F \in [0.5, 1]$, and the compromise tactic by $F=1$ or $F=0.5$. Let $v(is_{ijk})_{\text{old}}/\left| \lim_{ijk} - v(is_{ijk})_{\text{old}} \right|$, where $v(is_{n, k})$ is the value proposed by other party $a_{gn}$ to the issue is$_{ijk}$.

Alternatively, the factor $F$ can vary throughout the negotiation and be a function of a single criteria [1]. In this paper, we concentrate on the relative concession criteria. Let $v(is_{1, j})_{\text{new}}, v(is_{1, j})_{\text{old}}, \ldots, v(is_{n, k})_{\text{old}}, v(is_{n, k})_{\text{new}}$, be the values of is$_{ijk}$ successively offered by $a_{gi}$, with $V_{ijk}(v(is_{1, j})_{\text{old}}), V_{ijk}(v(is_{1, j})_{\text{old}})$, $\leq n$. Let $C_{total} = \left| v(is_{1, j})_{\text{old}} - v(is_{n, k})_{\text{old}} \right|$, be the total concession made by $a_{gi}$ on is$_{ijk}$ at a specific point in the negotiation. Let $C_{total} = \left| v(is_{1, j})_{\text{old}} - v(is_{n, k})_{\text{old}} \right|$, be the total concession made by $a_{gi}$ on is$_{ijk}$. We distinguish two functions for modelling $F$:

1. $F = 1 - \gamma e^{C_{total}/|C_{total}|}$, and
2. $F = \gamma e^{C_{total}/|C_{total}|}$

where $\gamma \in \mathbb{R}^+$. The five tactics are now defined by choosing different values for the parameter $\gamma$. For function (1), the stalemate tactic is defined by setting $\gamma \in [0, 1]$, the tough tactic by $\gamma \in [1, 0.5]$, the moderate tactic by setting $\gamma = 1$, the soft tactic by $\gamma \in [0, 1]$, and the compromise tactic by $\gamma = 0$ or $\gamma = \left| \lim_{ijk} - v(is_{n, k})_{\text{old}} \right| e^{C_{total}/|C_{total}|}$.

6. Related Work

Laasri et al. [3] describe a generic negotiation mechanism, but assume that agents are cooperative. Sycara [10] presents a negotiation mechanism that can be employed by non-cooperative agents and supports problem restructuring, but assumes the existence of a centralized mediator. Faratin et al. [1] present a negotiation mechanism that defines a number of negotiation tactics. The mechanism is rich, but no consideration was given to integrate it into a unified agent model.

We are interested in negotiation among self-interested agents. Our negotiation mechanism is generic and supports both dynamic constraint relaxation and problem restructuring. Also, our structure for representing negotiation problems allows the direct integration of planning and negotiation.

7. Discussion and Future Work

This article has introduced a negotiation mechanism and a set of negotiation tactics. There are several features of our work that should be highlighted. First, the structure of a negotiation problem defines the negotiation issues. Second, problem structure represents a clear link between the individual and social behavior of agents. Third, the mechanism supports constraint relaxation and problem restructuring ensuring a high degree of flexibility. Finally, the negotiation tactics support both cooperative and non-cooperative negotiation behaviors.

Our aim for the future is: (i) to define a set of negotiation strategies, and (ii) to validate experimentally the negotiation mechanism.

8. References