In "Applied Artificial Intelligence"

Special Issue on 'Socially Intelligent Agents'. 12, 1998

Personality Traits and Social Attitudes in Multi-Agent Cooperation

Cristiano Castelfranchi [1], *Fiorella de Rosis*[2], *Rino Falcone* [1] *and Sebastiano Pizzutilo*[2]

[1] Institute of Psychology, National Research Council, Rome, Italy

cris, falcone@pscs2.irmkant.rm.cnr.it

[2] Department of Informatics, University of Bari, Italy

derosis, pizzutilo@di.uniba.it

Abstract

In this paper, we discuss the meaning of personality and its role in socially intelligent multiagent systems. After examining the reasons behind the current trend towards endowing software agents with personality, we introduce our notion of personality as a combination of traits and attitudes. We characterise what we consider to be two basic elements of any cooperation activity (*delegation* and *help*) and we show how they can be diversified in relation to the agent's level of autonomy and cooperativity. We then describe how we formalise these forms of delegation and help, in GOLEM, a multiagent cooperation testbed, and we outline how these traits and attitudes can be organised into reasonable personalities and interesting interactive situations. Finally, we show how, in GOLEM, these traits and attitudes are involved in deciding what to do proactively or in response to other agents' social action, and in reasoning about other agents' mind.

proofs should please be sent to:

Fiorella de Rosis
Via Cairoli 101
00185 Roma, Italy

This paper is a revised version of a paper which was published in the Working Notes of the AAAI Fall Symposium on *Socially Intelligent Agents,* Technical Report FS-97-02, 1997.

In its present form, it has not been published elsewhere and has not been submitted for publication to other journals.

1. Introduction

Autonomy is a distinguishing feature of the agent paradigm: designing a 'supporting' agent equates to defining the circumstances in which the agent will activate itself, whether it will follow literally the received request or will try to understand what it is expected to do in the specific circumstance, to respond critically to this request, and so on. These design features may be seen in terms of 'type of delegation' given to the agent and 'type of help' that the agent is programmed to offer. Any software agent includes an implicit definition of the form of delegation it presumes to receive and of the form of help it is able to offer. In the majority of frameworks aimed at simulating multiagent systems' behaviour, every agent reasons only 'on itself' (that is, on what it can and wants to do) and applies for other agents' help only in case of need. Delegation originates from awareness of not being able to do by itself (Sichman et al, 1994) and request of help always concerns a well defined task. Symmetrically, offer of help is bound to the received request. Partial knowledge is contemplated in these systems, but not insincerity: agents are usually 'gullible', in that they are ready to believe to messages received and to retract their previous knowledge accordingly: see for instance (O'Hare, 1996). In the multiagent cooperation theory that we describe elsewhere (Falcone and Castelfranchi, 1997), the main assumption is that no 'optimal' delegation/help attitude can be defined, and that efficient multiagent systems may stem from a combination of different cooperation attitudes. We call 'personality' this combination of attitudes, consistently with a software anthropomorphization in which the program's knowledge base, control structure and input/output are seen as 'mental state', 'reasoning forms' and 'communication language'. We are convinced that a future society of artificial agents which will be built in a partially uncontrollable way will have to cope with heterogeneous and typically 'human' behaviours such as lie, elusion and similar. To test whether this world might, in some cases, be more efficient than a world in which more rigid and uniform behaviours are implemented, we designed a simulation framework that we called GOLEM: with this tool, we can formalise and simulate cooperation between agents with different kinds and levels of social delegation and help, represented as different social personality attitudes or traits.

In this paper, we first examine (in Section 2) how personality is being introduced into the the multiagent field; we then discuss (in Section 3) its meaning, also in contrast with other concepts (like emotions) which are frequently employed in the believable agents research. In Section 4, we introduce the GOLEM Project, by illustrating the cooperation personality traits that characterize its agents. Section 5 gives an example of simulation that can be performed with this testbed, and Section 6 describes its functions and its architecture. In Section 7, we illustrate, in particular, the various forms of abductive reasoning that GOLEM agents are able to perform, and discuss the crucial role of this form of reasoning in social intelligence. Section 8 gives a view into the future of this research, in which we plan to pass from stable personality traits to flexible social attitudes. Some final remarks conclude the paper.

2. Why do Agents need personalities?

Agents endowed with personalities or personality traits, characters, individual attitudes, etc. are spreading around recently, not only in entertainment and for artistic purposes (Hayes-Roth, 1995; Loyall and Bates, 1997) but also in interface and dialogue systems (de Rosis et al, 1996; Dryer, 1997), in theory of rational and reactive behaviour (Cohen and Levesque, 1990; Kinny et al, 1994), in social science simulation (Lomborg, 1994; Conte and Castelfranchi, 1995) and for Multiagent Systems validation (Cesta, Miceli and Rizzo, 1996). Which are the reasons of this trend? Just fashion, curiosity, novelty; or is this a necessary development of the "agentification" of AI? There are in fact several independent reasons for introducing personalities in Agents. Let's summarise them.

a. Social/cognitive Modelling

One of the major objectives of AI (and Alife) as science is modelling natural intelligence. Since in nature and in society agents have personalities and this seems an important construct in psychology, one might model personality, emotions or cognitive biases in agents to reproduce relevant features of human interaction.

b. Believability and interaction

Believability is one of the most important features for a natural user interaction with agents. In entertainment, it is mainly related to expressing emotions and caricatures (Reilly and Bates, 1995) and to reacting in a "typical", or "peculiar" way; thus, believable agents tend to have personality. Personalities were in fact first introduced in AI to make more "believable" and deceptive some systems like the paranoid PARRY. They now tend to be considered as part of friendly user interface design, after several studies showed that "people respond socially to computers and perceive them as having personalities" (Nass et al, 1995; Dryer, 1997).

c. Story and situation understanding

In making the required inferences for understanding a story or a situation, it is necessary not only to know the appropriate scripts and frames and the agents' intentions and beliefs, but also their personalities. The first quite complete and formal theory of personality was introduced exactly for this purpose (Carbonell, 1980). On the one hand, this perspective is strictly related to what is now called "believability". As Carbonell says: "Whenever a story includes character development of one of the actors, this development turns out to be useful and often crucial in formulating an understanding of the story. Here we deal with the most simple form of character development: the attribution of personality traits to actors in a simple story....knowledge about personality traits is necessary to understand the actions of the characters."(pp.217-19). On the other hand, this claim is connected to what is now called "agent modelling". In fact, not only in stories but also in real interactions (both in human or in virtual reality) "knowledge about personality traits is necessary to understand the actions" of the agents.

d. Agent Modelling

User stereoptypes and profiles proved to be useful in adaptive and cooperative human-machine interaction, to make correct ascriptions and abductions (Rich, 1989). The same is true in multiple agents' interaction (Castelfranchi and Falcone, 1996). We therefore need defining agents' classes and stereotypes, some of which are personality-based. For example: in user modelling, *student* denotes a role, whereas *aristocratic* or *thrifty* denote personalities (or, better, personality traits). Among agents, we might have classes like *mediator* or *executive agent* or *information filtering agent*, but also classes like *benevolent* or *self interested*, which in fact correspond to social personality traits or attitudes. In addition, an agent needs to have a model of the other agent in order to appropriately interact with it.

All these are interesting reasons for introducing personalities in the agents. However, we believe that there is some more principled reason that holds in the very basic philosophy of agent-based computing: its decentralised character, its open world assumption (Hewitt, 1991), its "experimental" approach.

e. Exploring and comparing strategies

One of the most interesting aspects of decentralised and MA systems is that they provide a scenario for experimental exploration of coordination mechanisms, behavioural strategies and organisational structures which could not be designed or predicted by centralized rational planning. This is crucial in the "open system" perspective that characterises the new AI of the '90ies (Bobrow, 1991; Hewitt, 1991). Exploring different behavioral, reactive or planning strategies in multiagent systems can be seen as exploring adaptivity, efficiency and coexistence of different "personalities" in agents' models. Personalities had already been implicitly introduced in the different kinds of commitment defined by Cohen and Levesque (1990) or by Rao and Georgeff (1991). The same was true in comparing different "forms" of rationality (Sichman et al, 1994) , etc. In MultiAgent worlds, no strategy can be defined as a-priori optimal, since the world is open, it changes, it is uncertain and unknown, and since other agents in the world will adopt strategies that might unexpectedly change the result of our actions. Strategies are good or bad in a given context, for a given population of interactive agents, with a given set of internal and external resources, with a given allocation of time and effort. Even if an optimal solution could be identified, it would be very complex to characterise it a-priori for different classes of situations. Thus, the new paradigm tends to be in favour of "experiments" and heterogeneity, where different possible solutions to a problem, different reactions to a situation, different ways of reasoning, different priorities in goals, are allowed to compete or coexist. Intelligence and efficiency tend to be seen (i) as emergent at the global level rather than being embedded in individual rules, (ii) as selected post hoc or (iii) as reduced to multiple dimensions and let coexist with less efficient strategies that adapt to changing situations. Heterogeneity is a very good explorative strategy and a very robust adaptive approach. Indeed, different solutions to a problem, different reactions to situations, different ways of reasoning, different priorities in goals, etc. are just "personalities". Thus, an agent-based approach to intelligence needs attaching personalities to agents.

f. Internal states and behaviour

Personality is strictly related to another very important and basic aspect of agents. We believe that one of the most important features of agents is that they have "internal states" (Shoham, 1993) and that their reaction to a given stimulus or their processing of a given input depend on this state. This seems to be one of the main differences between an "agent" and a software component, a module or a function. This implies that agents react in different ways or give different process results to the same input, depending on their internal state. So they have different reactive styles, either stable or transitory. Personality is just a specification, a sub-case of this general property. As argued in (Castelfranchi, 1995), this property (internal state mediation between input and output) is a very basic aspect of autonomy: autonomy relative to the stimulus, to the world. In agent-based computing, the introduction of personality will therefore be motivated also by the need to introduce different treatments of the same input or different processing reactions, that cannot be decided on the basis of external parameters and tests, or input conditions. These different "computations" are conditional to "internal" parameters which evolve independently of the agent sending the input and are unpredictable. Personalities are only an extreme of this feature: a stable set of (potentially transitory) internal states, acting and reacting modalities, forms of reasoning on the input. When these local and internal states and parameters cannot be reduced in terms of knowledge or ability (which can both be acquired), then they may be seen as "personality traits".

3. What is personality in agents

We call *personality trait* (Carbonell, 1980) any internal state or processing mechanism of the agents that: (i) differentiates a class of agents or an individual agents from other agents with which it is interacting or is compared; (ii) is relatively stable (either built in or inborn or learned, but now quite permanent) and cannot be just adopted or learned from outside on line; (iii) is mental: or mental attitudes (beliefs, goals, etc.) or mental styles[*]; (iv) has to do with motivations, with the way of choosing, of reasoning, of planning and so on. We agree with Carbonell that personalities are mainly goal based (see also Rizzo et al, 1997): some of them consist in taking a typical motivation or in assigning a special importance to a given goal (ex. *sadic*, *glutton*); others can be considered as implicit goals or preferences (see later). However, other personalities are rather based on "cognitive styles": ways of reasoning, attending, memorising, etc.

3.1. Personality traits and attitudes

Some personality traits are conditional on a given circumstance: they are just temporary *attitudes*[@]. An attitude is characterized by tests/conditions specifying the circumstance for its activation. An agent can assume an attitude or another (relatively to the same problem) depending on the circumstances or on its partners. Therefore, we distinguish in GOLEM between *traits* and *attitudes*: both are constituents of personalities. An agent can decide to change its attitude towards a given event, request, or agent, while it cannot do the same about its traits: these are not subject to contextual changes or decisions. That personality traits are stable does not mean that they are continuously relevant or active: if A is a *glutton*, when he is working this can be irrelevant. In presenting GOLEM personalities, we will first introduce some personality traits which are independent of situations or interactions. Later on, we will show how these traits could become more flexible social *attitudes*, by giving the agent the possibility of adopting them or not, depending (for example) on the partner's personality.

In short: *a personality is a consistent, believable, stable, and typical or distinctive cluster of traits and attitudes that are reflected in the agent's behaviour*. According to this definition, agents with different personalities must show different behaviours in similar circumstances; they should be consistent, by showing the same behaviours in not significantly different circumstances. Also doing this, they are "believable".

3.2. Personality and Emotions

Emotional states are part of the internal states that change an agent's cognitive process and reaction; they can also characterise the agent. Emotion-based personalities can be defined, like *shameful*, *fearful*, *pityful* and so

[*] A typical physical feature or a behavioural regularity is not a personality trait. It becomes a personality trait only if it is ascribed to (and viewed as the external expression of) some "internal" (mental) feature.

[@] Not in the sense of "mental attitudes" or "propositional attitudes" but in the common and psychological sense of "attitude" (disposition) towards another agent (or towards an event).

on: these personalities are characterised by the agent's propensity for a given emotional reaction. However, emotions and personalities should not be mixed up with each other, like it risks to happen in the "believable agent" domain. This is due to the fact that, in that domain, personalities are introduced just for the sake of "believability", and believability for sure requires emotional reactions (Elliott, 1994; Reilly and Bates, 1995; Hayes-Roth, 1995; Picard, 1996). In our view:

- emotions do not necessarily imply personalities, since there might be emotional behaviours that are shared by the whole population of agents and do not characterise particular agents or individuals;
- personalities are not necessarily related to emotions: they might be just based on (i) cognitive properties or styles, like a "fantasyful" or a "fanatic" agent, (ii) preferences and goals, (iii) interactive strategies (ex. Tit-for-Tat agents; or cheaters, etc.).

Of course, it is true that these cognitive styles, and in particular preferences and goals, can make a given type of agent (or individual) exceptionally liable to some emotions. However, these emotions are not the basis for constructing and characterising that agent, though being useful to recognise it. In addition, emotions are not necessary: agents might be free from emotions while having personalities.

4. The GOLEM Project

As we anticipated in the Introduction, the purpose of this Project is, on one side, to investigate how different cooperation attitudes may be combined in a socially intelligent multiagent system and, on the other side, how agents should be programmed, to show such a form of social intelligence. Long term objectives of GOLEM are therefore the following:

(i) to *provide an experimental evidence of* how different social attitudes perform, when they meet in a multiagent world;

(ii) to *identify especially interesting personalities* and situations, both for the theory of social interaction and for its application to software agents;

(iii) to *formalise non-benevolent social attitudes* in interaction and collaboration, with particular reference to opportunistic and exploitation behaviours and to deception.

In the immediate, we wish to *make explicit the mental state and the reasoning process* that motivate (and make possible) a given delegation or adoption strategy, and to *examine the internal and external compatibility* between these strategies: that is, which delegation - help strategies and attitudes may coexist within the same agent and in different agents in the same world.

In the next Sections, we will shortly introduce the theory of levels of delegation and levels of goal-adoption (help); we will then describe a set of basic delegation and help personality traits, how they combine into more complex "personalities" and interactive situations, how different agent's attitudes are related to each other and how the agents' personality traits influence the reasoning process.

4.1. Delegation, adoption, autonomy and deep cooperation

Delegation and Adoption are two basic ingredients of any collaboration and organization. The large majority of DAI and MA is based on the idea that *cooperation works through the allocation of some task of a given agent to another agent*, via some "request" (offer, proposal, announcement, etc.) meeting some "commitment" (bid, help, contract, adoption, etc). In other papers (Castelfranchi and Falcone, 1998; Falcone and Castelfranchi, 1997), an analytic theory of delegation and adoption was developed to contribute to understanding and clarifying the cooperative paradigm. Informally:

- *in delegation or reliance, an agent A needs an action of another agent B and includes it in its own plan*. In other words, *A is trying to achieve some of its goals through B's actions; thus A has the goal that B performs a given action.* A is constructing a MA plan (Kinny, 1994) and B has a share in this plan.
- *in adoption or help, an agent B has a goal since and until it is the goal of another agent A, i.e. B has the goal of performing an action since this action is included in A's plan.* So, also in this case B plays a part in this plan.

Both delegation and adoption may be unilateral: B may ignore A's delegation while A may ignore B's adoption. In both cases, A and B are, in fact, performing a MA plan.

One can distinguish among at least the following types of delegation: *(i) pure executive Vs open delegation; (ii) strict Vs weak delegation; (iii) delegation Vs non delegation of the control over the action; (iv) domain task Vs planning task delegation (meta-actions); (v) delegation to perform Vs delegation to delegate.*

The dimensions which characterize the *autonomy* of the delegated agent (B) from the delegating one (A) are the following : *(i) level of delegation 'openness', (ii) level of control of actions given up or delegated; (iii) level of decision left to B; (iv) level of dependence of B on A, as for the resources necessary for the task.*

The object of delegation can be specified minimally (*open delegation*), completely (*close delegation*) or at any intermediate level. We wish to stress that *open delegation* is not only due to A's preference, practical ignorance or limited ability. Of course, when A is delegating a task to B, he is always *depending on* B for that task (Sichman et al, 1994): he needs B's action for some of his goals. However, open delegation is also due to A's ignorance about the world and its dynamics: *fully specifying a task is often impossible or not convenient*. Open delegation is one of the bases of the *flexibility* of distributed and MA plans. In analogy with delegation, several levels of help can be characterized, that define the *level of collaboration* of the adopting agent: there are agents that help other agents by just doing what they were literally requested to do; there are agents that have initiative, have care of others' interests: they use their knowledge and intelligence to correct others' plans and requests that might be incomplete, wrong or self-defeating.

4.2. Cooperative personality traits in GOLEM

A personality in GOLEM is “a consistent, believable, stable, and typical or distinctive cluster of traits and attitudes that are reflected in the agent's behaviour”, but in the current implementation the traits just represent a consistent block of delegation and adoption strategies: these traits establish when to delegate or adopt a domain-action and which kind of help to provide. The delegation and adoption strategies that may characterize the widely differentiated agents' personalities vary along three main dimensions:

- intrinsic features of the agent who has to decide whether to delegate or to adopt a task (its capabilities and its inclination to jeopardize its own goals),
- features of the agent with which it interacts (its capabilities and its personality).
- features of the task.

Other attitude-dependent rules define criteria to assess when a 'goal conflict' exists and when to conclude that the other agent really needs help: these attitudes are less 'permanent' and can be assumed to evolve during interaction. We show, in **Figure 1**, some examples of personality traits introduced in GOLEM, that we will illustrate in the next Sections.

4.2.1. Delegation traits

Reasoning about delegation may end up in one of the following ways: (a) the intention to do a specific action by itself, (b) the desire to induce that intention on the other agent by delegating that action or (c) the decision to renounce to that action, by 'waiting'. Personality traits establish a "preference rate" among the three alternatives, in the form of strategies to decide which alternative to select. Some examples of these traits are shown in Figure 1.

FIGURE 1 ABOUT HERE

4.2.2. Helping traits and attitudes

Reasoning about adoption may end up with two alternatives: to help or to refuse helping. At least two personality traits influence making a choice between these two alternatives and establishing how to help: *level of propensity* towards helping and *level of engagement* in helping. As far as the first trait is concerned, various factors contribute to deciding whether to help: (a) own know-how, (b) presumed know-how of the delegating agent and (c) compatibility of the required action with own goals. Personality traits establish a priority for each of these factors, as shown in the examples in Figure 1. As far as the level of engagement in adoption is concerned, the helper may follow the received delegation literally, or it can interpret it extensively, according to how much it really wants to meet the delegating agent's desires; some typical help levels are shown, again, in Figure 1. **Figure 2** exemplifies these help levels, in a domain in which two types of blocks (big ones and small ones) coexist, and in which two agents (that we call Adam and Eve) cooperate in building several types of block structures.

FIGURE 2 ABOUT HERE

Secondary to the described personality traits, other attitudes further diversify the behaviour of the helping agent:

- *control of conflicts* between the requested action and its own goals: a delegated action can immediately bring to a state which is in conflict with the helper's goal state (a situation of 'surface-conflict'); in other cases, though not producing an immediate conflict, it can be part of a delegating agent's plan which, in the long term, will produce a conflict (a 'deep-conflict'). A *deep-conflict-checker* will check that no such conflicts are created by the requested action: to this aim, it will make a plan-recognition on the delegating agent's mental state. A *surface-conflict-checker* will, instead, restrict itself to examining the immediate consequences of the requested action.

- *control of the delegating agent's know-how*: this, again, can be restricted to examining whether that agent would be able to perform the requested action (in a *surface-knowhow-checker*) or can go deeper to examining whether alternative plans exist, which bring to the delegating agent's presumed goal and that this agent would be able to perform by itself (in a *deep-knowhow-checker*).

The difference between these attitudes and the main traits which characterize the behaviour of the helping agent is in their 'temporary' and 'relative' nature. For example: an helper may adopt a *deep-conflict-checker* attitude in a specific turn or towards a specific agent, possibly as a result of previous interactions with that agent.

4.3. Consistent personality traits

Agents' personalities are defined as a combination of delegation, helping and reaction traits and attitudes: this corresponds to the well know stereotype-based approach to modeling, in which multiple inheritance is exploited to produce a multi-faceted representation of a user or an agent (Rich, 1989). However, not all combinations of attitudes and traits produce a consistent personality profile. Some of them may be inconsistent as for the agent's rationality and efficiency, or may produce an *unplausible* (unbelievable) character. Of course, the two perspectives are not overlapping. Several "irrational" combinations are perfectly believable and correspond to interesting, antieconomic personalities: we consider them in GOLEM not only because of their believability, but also because in several applications (for example, interfaces) selfish efficiency of agents is not relevant.

Figure 3 enumerates the personalities produced by combining the delegation and helping traits that are described in Figure 1.

FIGURE 3 ABOUT HERE

Personalities 11 or 16 (and 12 and 17), for example, are not very rational: an agent like this never delegates, or delegates only if needed; that is, it never (or only when necessary) uses the resources/abilities of other agents; however, it is, at the same time, ready to waste its own resources for the others, even if they are lazy or hanger-on. Such an agent would be doomed to be hardly exploited by the others, but is not necessarily unbelievable. Cases 11 and 12 are examples of plausible personalities: they represent a very kind and polite agent that doesn’t like to disturb the others and prefers to meet by itself its own needs; at the same time, he is ready to help the others. Of course, these are not the only believable combinations for a *delegating-if-needed* agent. Combination 13 is believable, as well: it represents an agent that likes equity, symmetry and autonomy, which delegates only when necessary and helps only agents that delegate when necessary; if it can do a task by itself, it does it, and believes that the others should do the same. Remaining combinations in the same row are believable too, although they produce quite different characters. Combination 15, for example, describes quite an antisocial agent, though not so extreme like the 20: it doesn't like to have social relations, it refuses to help the others, and asks something to others only when it is obliged to do so.

If we consider the *lazy* row, we see two very implausible personalities (1 and 2): why should a *lazy* (which always tries to do the minimum) adopt altruistically the goals of others? This row shows also some believable combinations; 4 and 5 are quite consistent: these agents try to exploit the others as far as they can; they always try to delegate and either refuse to help, or help only when the task is useful for their goals. They are, thus, very egoist. Combination 3 (Mister "if I really have to do it...!!") is an interesting character as well: it bases its behaviour on the coherent principle of "doing only by necessity"; it does the task by itself when it cannot delegate it, and helps only when the others cannot do the task by themselfs.

In conclusion: different combinations of the same delegation and helping personality traits generate several believable personalities, with their own consistency; others are implausible and inconsistent; others are not very rational from an economic point of view.

4.4. Interesting encounters

When different agents are created in GOLEM, they should be given personality traits that produce social interactions interesting to investigate. For example: if two *never-delegating&selfish* agents meet, no delegation and no help will occur; if two *hanger-on&selfish* agents meet, no cooperation attempt will succeed; if an *hanger-on* meets a *hypercooperative,* there will be complete exploitation. These interactions are not particularly exciting if only a couple of agents is examined; they may be interesting in larger mixed populations (Cesta, Miceli and Rizzo, 1996).

Among the many possible encounters, we selected (in the present prototype) a few combinations that we consider to be especially interesting to study by experimenting their coexistence. For example: the interaction between a *benevolent* & *deep or surface checker* agent, or a *supplier,* and a *delegating-by-need (*see the example in the next Section). As we will better illustrate in Section 8, what we consider to be especially interesting, in letting delegation and help personalities to meet, is to enable them to vary and adapt to each other. The most interesting social personalities and interactions are those in which agents do not behave in a fix and rigid way, independently of the personality of the agent with which they are interacting, but in which they tune their social interaction to the personality of the other. In other terms, interactions between socially different agents will be more interesting when stable personality traits will be transformed into context-dependent "attitudes".

5. An example

To give some idea of the kind of simulation we wish to implement in GOLEM, let us look at the following example (**Figure 4)**.

FIGURE 4 ABOUT HERE

Adam and Eve are two agents playing in a blocks world in which small or big blocks can be handled. They have to play in turn, and can make just one domain-action at each turn. Eve's goal is to build a twin-tower (that is, to reach a situation in which two towers, of small and big blocks, coexist); she can handle just small blocks; she doesn't know Adam's goal, but believes (correctly) that he can handle big blocks as well as small ones. Adam's goal is to build a bell-tower (a big tower with a small block on top of it): he doesn't know Eve's goal but has a correct image of her capabilities.

turn 1: **Eve** plans how to reach her goal: she has to build a small and a big tower; she might do the small tower, but is *lazy*, and therefore decides to request to Adam to do it.

turn 2: **Adam** is *a benevolent*, and will help Eve if the requested action does not conflict with his goal. His plan is to build a big tower, and to then place a small block on top of it; Eve's request is therefore in conflict with his own goals; he then refuses to help her, and goes on with his plan by building a big tower.

turn 3: **Eve** completes her plan by building a small tower and wins: the game ends up because its rules establish that a reached goal cannot be destroyed. Adam lost essentially because of his benevolent and surface-conflict-checking personality.

6. Outline of GOLEM

Our testbed was designed according to the following criteria:

a. *domain independence:* agents are built separately from the application domain, and their behaviour is described in a domain-independent way;

b. *flexibility in the description of agents behaviour*: decision strategies that agents apply in the different phases of cooperation can be revised easily; new individual personalities can be added if needed, and new combinations of personalities, for groups of interacting agents; new relations between personality and reasoning can be introduced, new inter-agent communication forms, and so on;

c. *flexibility in the representation of mutual knowledge*: an agent may have an incomplete or even incorrect knowledge of other agents, whereas it knows exactly itself,

d. *flexibility in the levels of sincerity in communication*; one of the personality traits that can be introduced in an agent description is its 'propensity to lie or to be reticent': this affects the way that an agent's decision is transformed into a communicative act and the way that communicative acts are interpreted by agents.

Opposite to these abstraction and flexibility features in the agent definition, we made two simplifying assumptions about the system functioning:

e. *we limit to two the number of interacting agents;*

f. *we serialize their activity*, with a synchronous 'turn taking' behaviour.

We plan, however, to relax these two assumptions in the next release of GOLEM.

6.1. Functions

The testbed includes the three main functions that are typical of these systems (Decker, 1996). The first two of them enable building a 'world' by defining the two agents characteristics and the domain in which they will play: a graphical interface guides the user in this description and an interpreter translates it in an internal form, by checking syntax errors. The third function enables simulating a game play, starting from an initial state of the world.

6.1.1 Domain facility

An application domain is represented as an oriented graph, whose nodes correspond to domain-states and whose arcs correspond to domain-actions bringing from one state to the next. Nodes are objects to which we associate a state *name* s, a symbolic *description* of the state and the name of an *image* which is employed to represent it on the interface. Arcs are objects to which we associate an action-name a.

This description enables computing the values of some properties of a state s:

(Performable a): "a can be performed in s";

(Achieve a s'): "a, performed in s, brings to s' ";

(Conflicts a s'): "a brings, from s, to a state which is in conflict with s'"; two states are 'in conflict' when their descriptions (with the addition of some 'frame condition') are contradictory.

In the blocks world in Figure 4, (Ab, Bb, Cb, Db) are the names of the big blocks and (as, bs, cs, ds) are the names of the small ones; stock, big-building, big-tower and so on are state names in this domain; make-big-tower, Stack-s(x y) and so on are action names.

An example of conflict between states: big-tower is in conflict with bell-tower by virtue of the frame condition: (On x y) -> not (Clear y).

This domain description enables us to implement plan-evaluation and goal-recognition functions as algorithms of path-searching in a graph.

6.1.2. Agent development facility

Our rational agents have a *mental state* which includes: (i) a *general knowledge* about the way that personalities affect reasoning and (ii) a *specific knowledge* about themselfs and about the other agent. Knowledge about own mental state is correct, complete and consistent; the image of the other agent may be incomplete or uncertain. The basic constituents of a mental state are the following:

- a set of *private and communicative actions* that agents can perform;
- a set of *reasoning and commitment rules* which settle the agents behaviour in the various phases of the play;
- a set of *personality* traits;
- a set of *basic belief*s;
- a *domain-goal* (domain-state that the agent desires achieving).

Private and communicative actions are the same for all agents, as well as commitment rules. Agents differ in the personality traits (and consequently, as we will see, in the reasoning rules), in the basic beliefs and in the domain goals.

a. actions

Like in (O'Hare, 1996), we classify private actions into physical and cognitive ones.

Physical actions correspond to domain transformations or control activities:

(Perform Ai a): "Ai performs a domain-action a" ,

(WaitUntil Ai a): "Ai controls the domain state, to verify that a has been performed".

Cognitive actions correspond to forms of reasoning. For instance:

- *infer beliefs*: apply resolution-based reasoning to infer whether a particular belief is the logical consequence of an agent's mental state.

- *goal recognition*: given a general domain knowledge, a 'history of interaction' and some knowledge of the other agent's mental state, abduce its domain goal;
- *plan evaluation*: given a general domain knowledge and given a present state s and a goal state g, select a 'reasonable plan' that enables achieving g by responding to some optimality criterion.
- *cognitive diagnosis* (abductive reasoning about another agent's mental state); given: (i) a general knowledge about the way that personalities affect reasoning, (ii) a communicative or physical action that was performed by the other agent, (iii) a prior knowledge about the mental state of that agent and (iv) a history of interaction, revise the image of that agent's mental state (its personality traits, its ability to perform domain-actions and its beliefs about other agents' abilities and intentions);
- *ATMS-based updating* of the other agent's mental state: update personality traits, abilities and intentions of an agent so as to ensure consistency in this image.

Communicative actions are speech acts about the agents' abilities and intentions: Request, various types of Inform and Query.

b. rules

We model separately the intention forming process from translation of intentions into a -private or communicative- action: we claim that the first process is personality-dependent, whereas the second is not. Within this distinction, we further classify rules according to the cooperation phase to which they apply, by distinguishing among delegation, help and reaction. We then have four types of rules overall: delegation/help reasoning and delegation/help commitment. As reasoning by an agent is aimed at deciding 'what to do in the present turn', we do not need representing time in our language. This means that, in GOLEM, agents cannot intend to perform a specific action in a specific time instant like, for instance, in (Shoham, 1993) or in (O'Hare, 1996): they can only decide whether or not to do it *at their next turn*. A couple of agents is programmed in two steps:

Step 1: building the *general knowledge* component:

1. defining the general rules in the world, that is all reasoning and commitment rules that might be included in the mental state of any agent;
2. defining a list of personality traits;
3. mapping personality traits into reasoning rules;

Step 2: building *two specific agents,* Adam and Eve; this step now requires only assigning to Adam (and to Eve) a set of personality traits and a set of basic beliefs and goals. The mental state of the two agents (their reasoning and commitment rules) will be built automatically from the personality trait table. **Figure 5** gives a semi-formal definition of rules.

FIGURE 5 ABOUT HERE

Figure 6 shows some of the reasoning rules of Adam and Eve, in the example of Figure 4.

FIGURE 6 ABOUT HERE

----

c. personalities

A personality trait is modeled as a logically and cognitively consistent combination of reasoning rules. As we have seen in the previous Sections, different traits affect delegation and help traits and attitudes: an agent's personality is therefore described by a plausible combination of a delegation and an help attitude, and a set of reasoning rules that correspond to these traits is attributed to the agent's mental state, accordingly.

d. basic beliefs

these are ground belief and goal atoms which represent the agent's knowledge about itself and about other agents; they are (obvioulsy) agent and domain-dependent. The following are examples of Adam's basic beliefs and goals in Figure 4:

(GOAL Adam (T bell-tower)), (BEL Adam (CanDo Adam make-big-tower)),
(BEL Adam (CanDo Adam make-small-tower)),
(BEL Adam (CanDo Eve make-small-tower)), (BEL Adam (GOAL Eve (T twin-tower))).

6.1.3 Evaluation facility

Our agents play in a domain, by trying to achieve their -compatible or conflicting- goals. The user can set the conditions of a simulation and follow how it proceeds, through a graphical interface. A particular simulation starts by selecting a 'world' (that is, a couple of agents with defined mental states) and by setting the initial domain state and the agent which 'moves first'. The two agents introduce themselves by declaring their personalities and abilities; in this introduction, they may give partially incorrect or 'abstract' descriptions, or may even lie about them. For example: Eve might omit the description of her abilities, might say that she is able to 'make towers' without specifying whether they are big or small, or might tell that she is not able to make small towers whereas she can do them. She might be vague, as well, in describing her personality; for instance, she might say "I'm someone who tends to delegate" rather than specifying whether she is *lazy* or *hanger-on*. This introduction of Eve initializes Adam's image of Eve's mental state. In any phase of interaction, users can look at the agents' mental state in a graphical *agents window*. They can permanently follow the progress of the play by means of a *domain window* (which represents graphically how the domain state evolves during the game play) and a *dialog window* (which shows the communicative and physical actions the two agents perform).

6.3. Architecture

Agents of GOLEM do not interact directly. They exchange messages through a *Message Board*, which holds knowledge about the domain state and how domain states can be modified. A *Message Board Handler* responds to agent queries about this knowledge; it also receives "(DO (action))" commitments at the end of each turn: it changes the domain state according to physical actions, forwards communicative actions to the *Interface Handler* and sends back a "(DONE (action))" message to the agent that moves next. The Interface Handler is responsible for user interaction: it displays graphically the domain state and the agents' characteristics, and translates communicative actions into natural language sentences.

We make a distinction between two levels of agent programming:

- *low-level* reasoning followed in deciding whether to delegate or help and how to react to a refusal of help is represented declaratively, in the rules;
- *high level* reasoning followed in a turn is represented by a procedural knowledge.

This allows a flexible representation of decision strategies in delegation, help and reaction, with their links to personality traits. On the contrary, at present we consider the high-level reasoning cycle to be more stable, the same for all agents and therefore personality-independent.

Some cognitive actions (infer beliefs, goal recognition and plan evaluation) are implemented in Lisp; the interpreter of the agent programming language is in YACC and C++; all other modules (including the Interface and the Message Board Handlers) are in Java.

7. Personality-based social reasoning

We described, in Section 4.2, how personality affects the decision on whether and how to delegate, to help and to react. If we now reconsider the example in Figure 4, we can notice that the game result changes when the two agents' personalities are changed, even though their abilities and the domain conditions are left invaried. For instance: Adam is defeated in the game because he is a *benevolent & a surface conflict-checker*; in the same conditions, a *deep-conflict-checker* would discover the conflict with Eve's goal and would pass the hand without doing any domain action; he would not win either in this case, but at least the game would be quits.

This example also shows that introducing a new personality trait into an agent's mental state may require expanding its reasoning abilities. In particular, abduction on the other agent's mental state is, may be, the form of reasoning which best characterizes socially intelligent behaviour: in order to socially interact with others, an agent needs some representation of their goals, intentions, knowledge and know-how. This can be based on personal acquaintance, on memory of interactions (Lomborg, 1994), on reputation (Conte and Castelfranchi, 1995) or on self-presentation (Sichman et al, 1994). It can be inherited, as well, by the class to which the agent is known or presumed to belong or can, finally, be abduced from the other agent's practical and communicative behaviour (de Rosis and Grasso, 1997). As we mentioned before, in GOLEM the agents self-description at the beginning of the play can be partial or fuzzy but is always consistent. An example: Eve can describe herself as a "*lazy* person who is able to handle blocks of small dimensions and would like to build a twin tower". More generically, she might introduce herself as "someone who tends to delegate tasks and likes complex structures". In this second formulation, it will not be clear to Adam whether Eve is *lazy* or just *hanger-on* and whether her domain-goal is a twin-tower or some state in which several structures coexist. The agent can also omit personality traits from this description. As a sincere-assertion assumption underlays the system, Adam acquires the described features to build up a first image of Eve, and eventually updates this image during interaction by applying several forms of abductive reasoning.

We now examine in more detail some examples of cognitive diagnosis that GOLEM agents are able to perform: the rules in this Section should not be considered as agent programming rules (such as, for instance, those shown in Figure 6), but as examples of agents' behaviour that is produced by abductive reasoning.

a. Abducing the other agent's plans and goals

Plans and goals can be inferred by reasoning on the domain knowledge; this is needed in several cases. Let us consider, again, the example in Figure 4:

If Eve requests Adam to make a big tower,
and Adam doesn't know anything about her goals
and he is a *deep-conflict-checker,*
then he will try to infer Eve's goal;
if he infers that her goal is just big tower,
then he will understand that he lost
(because big-tower is in conflict with bell-tower);
if he infers that she wants to come to have a bell-tower,
then he will conclude that they are in a cooperation situation.

This form of reasoning is also needed by other personalities, like *hypercooperatives* or *critical-helpers*.

b. Abducing the other agent's personality

Our agents can apply abductive reasoning for inferring, as well, the personality of their partner from its behaviour. Let us examine some cases, again from our initial example.

- *abducing delegation personalities*

When Eve requests Adam to make a small tower
then he will infer that she might be a *lazy,* a*delegating-if needed* or an *hanger-on.*
If later on, she makes the small tower,
then he will revise this belief, to exclude that she is a*delegating-if-needed.*

- *abducing help personalities*

When Adam refuses to make the small tower,
then Eve may infer that either he is not a *hyper-cooperative*
or he is not able to perform that action.

c. Abducing from the other agent's personality

Knowing the personality of another agent provides some hints on its abilities and on its second-order beliefs. For instance:

If Eve asks Adam to make a small tower,
and he believes that she is not able to do it by herself and that
she made the request because she is a *delegating-if-needed,* and
Adam refuses to help her just because he wants to avoid that she wins,
and, in a later turn, Eve performs the small tower herself
then Adam will revise his image of Eve's abilities (and his strategy, as well!).
If at turn 3 in Figure 4, Eve knows that Adam is a *benevolent*,
then she may infer that either he is a *supplier* or
he believes that his goal is in conflict with her goal.

This knowledge might also be employed by agents to decide whether to delegate or not (and at which level) and whether to help or not (and at which level), based not only on their own personality but also on the personality of the other agent. Some examples of how to *infer willingness and other obstacles or opportunities for delegation* from the other agent's personality:

If Eve believes that Adam is a *supplier*
and that he knows that she is able to make a small tower,
then she will not delegate him this action;

If Eve believes that Adam is a *supplier* and a *surface-conflict-checker*,
that he does not know that she is able to make a small tower and
that there are no conflicts between that action and Adam's goals
then she will delegate him that action;

If Eve believes that Adam is a *benevolent* and that
there is a conflict between her plan and his plan,
then she will not delegate him that action.

If Eve knows that Adam is a *benevolent* and that
there is a conflict between her plan and his plan,
but she also knows that the conflict is deep, whereas
Adam is just a *surface-conflict-checker*,
then she will delegate him that action (and cheat Adam).

d. Personality-based abduction

In absence of other information, the selection of the 'most plausible' hypothesis, in plan recognition or cognitive diagnosis, can be guided by attitudes which originate from the relationship between the reasoning agent and its partner. Two examples:

If Eve requests Adam to make a small tower and
Adam is *highly cooperative* and has no information about Eve' goals and,
from his plan recognition process,
he can make two hypotheses about Eve's final goal-state,
one of which (twin tower) is in conflict with his own goals,
whereas the other (bell-tower) is not, and Adam is *suspicious*
then he will select the hypothesis of conflict and will not help Eve.

If Eve requests Adam to make a small tower and Adam is a *supplier* and
has no information about Eve's know-how and personality and,
from his cognitive diagnosis process,
he can make two hypotheses about the reasons behind Eve's delegation,
one of which being that she could do the requested action but is *lazy*,
the other that she cannot do it and is a *delegating-if-needed*,
and Adam is *trustful*
then he will select the second hypothesis and will help Eve.

These examples show that new personality attitudes affect this aspect of reasoning: being *suspicious* or *trustful* is unlikely to be a permanent trait, but may rather be induced by the other agent's previous behaviour.

8. Future developments: considering the other agent's personality in delegation and help.

The main future development of GOLEM is to *pass from stable personality traits to flexible social attitudes* and strategic interactions. We define an attitude as "a trait conditional to circumstance": in this perspective, an agent is not permanently an *hanger-on* or a *supplier*, but assumes such an attitude depending on the personality or the attitude of the agent with which it interacts. Two problems arise from this new perspective:

a. which are the interesting and believable interactions between attitudes? i.e. which delegating attitude should deal with which helping attitude and viceversa? Some examples of rules for activating the delegating attitude:

IF the other agent is *hypercooperative* or *benevolent,*
then be *lazy*
IF the other agent is a *supplier*,
then be a *delegating-if-needed*
IF the other agent is a *non-helper*,
then be a *never-delegating*

Similar rules can be proposed for activating helping actitudes. These rules are the *reactive* counterpart of some of the behaviour rules that we exemplified in Section 7c: agents might establish their delegation attitude after a throughout abductive reasoning on the other agent's mind (by *inferring obstacles or opportunities for delegation*) or might be programmed to behave in a reactive way, by just applying some attitude-activating rules. In this case, the same attitude may be activated by different rules and the same condition may activate different attitudes; in a given circumstance, several rules may be applied, and just one of them will be selected by a given personality: as a result, not all the agents will behave in the same way in the same situation.

b. what, then, becomes a personality?

A personality is a believable cluster of attitude-activating rules. While an agent assumes a given helping attitude in front of an *hanger-on*, another agent assumes a different one. While an agent is helping only certain kinds of agents, another helps in different circumstances. We plan to study these more sophisticated characters, to define believable, and consistent combinations of flexible social attitudes. In this new view, a rigid personality (like a *never-helper*) will be just a very special case of fixed/constant attitude. Notice that flexibility will make reasoning about the other agent much more complex, and interaction much more strategic. The two agents will not have stable traits but context-dependent attitudes: so, the attitude of an agent in a given turn will depend on the attitude assumed by the other agent in the previous turns. An agent will not have just to know which is the stable helping trait of the other but, more refinedly, which helping attitude it might take in response to a given delegating attitude, ... and even more than this. In a satisfactory model of social interaction, the helping or delegating attitude of an agent should not depend only on the *complementary* attitude of the other: it should depend, as well, on the *corresponding* attitude of the other. An agent will help or not depending not only on the kind of delegation of the other, but also on its helping attitude; for example: never help a *non-helper*.

9. Conclusion

After examining the reasons behind the current trend towards endowing software agents with personalities, we introduced our notion of personality as a combination of traits and attitudes. We defined the social personality traits and attitudes that affect help and delegation in GOLEM and how they are combined into reasonable and socially interesting interactive situations. We showed how, in GOLEM, these personality traits are involved in deciding what to do proactively or in response to the other agent's actions, and in abductive reasoning about the other's mind. Finally, we argued that, as social action is strongly affected by the personalities of interacting agents, this feature should be part of "agent modelling"; in other words, it is reasonable to delegate a given level of task to a given kind of helping agent, and it is reasonable to give/offer a given level of help to a given kind of delegating agent. In future developments of GOLEM, we will provide experimental evidence of these statements, by exhibiting results of simulations about adaptivity of different personalities in different interactions and by evaluating believability of agents' personalities and interactive exchanges.

New interesting and useful social personality traits need to be included in GOLEM. For example, personalities based on typical goals (in the blocks world, "agents that always want to build some kind of tower") or on propensity to deceive. Especially important is the introduction of an "exchange" personality, that is an agent that proposes an "exchange", either spontaneously or in response to some delegation : "I will do that for you if you will do this for me". Of course, we are working also to a better systematisation of traits and attitudes in an inheritance hierarchy, in reasonable complex personalities and in multiagent situations relevant for both the theory and the application of cooperative systems.

References

Bobrow D. 1991. Dimensions of Interaction, AI Magazine, 12, 3, 64-80.

Carbonell J. 1980. Towards a process model of human personality traits. Artificial Intelligence, 15.

Castelfranchi C. 1995. Guaranties for Autonomy in Cognitive Agents Architecture. In: M Wooldridge and N Jennings (Eds). Agent Theories, Architectures and Languages. Springer Verlag, Heidelberg.

Castelfranchi C. & Falcone R. Towards a Theory of Delegation for Agent-based Systems, Robotics and Autonomous Systems, Special issue on Multi-Agent Rationality, Elsevier Editor, (in press).

Cesta A., Miceli M. and Rizzo P. 1996. Effects of different interaction attitudes on a multi-agent system performamce. In W. Van de Welde & J W Perram (Eds.), Agents breaking away. Springer-Verlag.

Cohen P. H. and Levesque H. J. 1990. Rational interaction as the basis for communication. in P R Cohen, J Morgan and M E Pollack (Eds): Intentions in Communication. The MIT Press.

Conte R. and Castelfranchi C. 1995. A simulative understanding of norm functionalities in social groups. In: R. Conte e N. Gilbert (Eds.) Artificial societies: The computer simulation of social life. UCL Press.

Decker K. S. 1996. Distributed artificial intelligence testbeds. In: Foundations of Distributed AI, (G M P O'Hare and N R Jennings Eds), John Wiley & Sons.

de Rosis F., Grasso F., Castelfranchi C., Poggi I. 1996. Modeling conflict resolution dialogs between believable aents. ECAI Workshop on Conflicts in AI.

de Rosis F. and Grasso F. 1997. Simulating plausible conflict-resolution dialogs. Submitted to the 1rst Workshop on Human-Computer Conversations. Bellagio.

Dryer D.C. 1997. Ghosts in the machines: personalities for socially adroit software agents. AAAI Symposium on Socially Intelligent Agents, AAAI Press Technical Report FS-97-02, American Association for Artificial Intelligence.

Elliott C. 1994. Research problems in the use of a shallow Artificial Intelligence model of personality and emotions. Proceedings of the 12° AAAI.

Falcone R. & Castelfranchi C. 1997. "On behalf of ..": levels ofhelp, levels of delegation and their conflicts, *4th ModelAge Workshop*:"Formal Model of Agents", Certosa di Pontignano (Siena).

Hayes-Roth B. 1995. Agents on stage: advancing the state of the art of AI. Proceedings of IJCAI95.

Hewitt C. 1991. Open information systems semantics for distributed artificial intelligence. Artificial Intelligence 47,79-116.

Kinny D., Ljungberg M. , Rao A., Sonenberg E., Tidhar G. and Werner E. 1994. Planned Team Activity, in C. Castelfranchi and E Werner (Eds.) Artificial Social Systems, Springer-Verlag LNAI 830.

Lomborg B. 1994. Game Theory vs. Multiple Agents: The Iterated Prisoner's Dilemma. In C. Castelfranchi and E Werner (Eds.) Artificial Social Systems - MAAMAW'92, Springer-Verlag LNAI 830, 69-93.

Loyall A. B. and Bates J. 1997. Personality-Rich Believable Agents That Use Language. In Proceedings of Autonomous Agents 97, Marina Del Rey, Cal.,106-13.

Nass C., Moon Y., Fogg B. J., Reeves B. and Dryer D. C. 1995. Can computer personalities be human personalities? International Journal of Human-Computer Studies, 43, 223-239.

O'Hare G. M. P. 1996. Agent Factory: an environment for the fabrication of multiagent systems. In the same book.

Picard R. 1996. Does HAL cry digital tears? Emotion and computers.

Rao A. S. and Georgeff M. P. 1991. Modeling rational agents within a BDI-architecture. In Principles of Knowledge Representation and Reasoning.

Reilly W. S. and Bates J. 1995. Natural negotiation for believable agents.Carnegie Mellon University CMU-CS-95-164.

Rich E. 1989. Stereotypes and user modeling. In: User Models in Dialog Systems. A Kobsa and W Wahlster (Eds), Springer-Verlag.

Rizzo P., Veloso M.V., Miceli M., Cesta A. 1997. Personality-Driven Social Behaviors in Believable Agents. AAAI Symposium on Socially Intelligent Agents, TR FS-97-02.

Shoham Y. 1993. Agent-oriented programming. Artificial Intelligence, 60.

Sichman J., Conte R., Castelfranchi C. and Demazeau Y. 1994. A social reasoning mechanism based on dependence networks. In Proceedings of the11th ECAI.

Walker M. A., Cahn J. E. and Whittaker S. J. 1997. Improvising Limguistic Style: Social and Affective Bases of Agent Personality. In Proceedings of Autonomous Agents 97, Marina Del Rey, Cal., 96-105.