The Great Deceivers: Virtual Agents and Believable Lies

João Dias (joao.dias@gaips.inesc-id.pt)
INESC-ID and IST, Universidade Técnica de Lisboa, Portugal

Ruth Aylett (r.s.aylett@hw.ac.uk)
Heriot-Watt University, Edinburgh, Scotland

Henrique Reis and Ana Paiva (henrique.reis@ist.utl.pt, ana.paiva@inesc-id.pt)
INESC-ID and IST, Universidade Técnica de Lisboa, Portugal

Abstract

This paper proposes a model giving Theory of Mind (ToM) capabilities to artificial agents to allow them to carry out deceptive behaviours. It describes a model supporting an N-level Theory of Mind and reports a study to assess whether equipping agents with a two-level ToM results in them being perceived as more socially intelligent than agents with a single-level ToM. A deception game being developed for intercultural training of children, used for this study, is described. Finally, we report results from this study consistent with the hypothesis that a two-level Theory of Mind better supports agents in deceptive behaviour.

Keywords: Virtual Agents; Theory of Mind; Deception

Introduction

The work reported in this paper arises from the use of synthetic graphical characters interacting in rich virtual worlds. These may be required for interactive drama applications (Mateas & Stern, 2003), or for story-based education and training applications (Paiva et al., 2004) (Swartout et al., 2006). A key criterion for success is that such agents be *believable*, that is lead a user, or viewer, to feel that they have an inner life of their own, with goals, motivations and emotions, and are in some sense 'alive' (Bates, 1994). Thus interaction between such characters must display features related to human-human interaction; whether the actions they carry out, their emotional expressions, ability to exhibit empathy, or non-verbal as well as verbal communications. Such features must be contextually appropriate, and in order to achieve this, characters may be driven by an architecture uniting cognitive and affective models, for example using a cognitive appraisal approach (Dias & Paiva, 2005) (Marsella & Gratch, 2009).

Computationally implemented cognitive appraisal models are often naive, assuming entirely open behaviour, sometimes referred to as meeting the *sincerity condition* (Searle, 1976). However, this is unusual in everyday human-human communication where deception often occurs. This may be as simple as masking anger in front of a social superior or fear in front of a child on a dark night (Rosis, Pelachaud, Poggi, Carofiglio, & Carolis, 2003), (Prendinger & Ishizuka, 2001), or as complex as deliberately misleading or lying to another person in order to gain an advantage. Deceptive behaviour includes not only the generation of false beliefs in others but also the claiming of desired identities, the exchange of non-existent emotions, and the communication of false preferences or opinions (Wyer & Epstein, 1996). Thus deception can be seen as a human-like characteristic that would enhance the believability of synthetic characters portrayed in real world social situations.

A Theory of Mind (ToM) process allows an agent to attribute an artificial mental state to another agent and reason about it. In a single-level ToM, agent A can represent only its belief about what an agent B is thinking; an agent C that can not only model what B is thinking but can also model what agent B thinks about C has a two-level ToM. In this paper we investigate the hypothesis that an agent with a single-level ToM will be less successful in believable deception then an agent with a two-level ToM. Deception cannot be investigated in abstract but requires a concrete scenario. Our work uses an interactive game played by and with autonomous graphical characters. This is based on the popular game Mafia, or Werewolf, described below, in which deception is fundamental to successful play. The characters are implemented with a cognitive appraisal-based architecture (Dias & Paiva, 2005) that includes a deliberative mechanism and has been extended to support an N-level ToM mechanism.

Background and Related Work

We define a "lie" as a direct communicative act that an agent performs to deceive another agent. We consider deception through verbal mechanisms - speech acts - though deception may also be achieved through non-verbal mechanisms. Deception has been widely studied in AI, though usually with disembodied software agents.

GOLEM (Castelfranchi & deRosis, 1998) is based on the blocks world of AI planning research. Goals conflict, since agents aim to build different structures from the same available blocks. Agents can achieve goals through their own actions or by asking for "help" from others. Agents have task delegation and adoption preferences and different capabilities, used to plan their actions based on their knowledge of other agents. Deception is instrumental, resulting only from goal conflicts, though it extends to deception about capabilities, goals or personality. However, agents in GOLEM can only produce lies within this limited scope. They cannot for example lie about the requests they have made or plan to make. This would require second order reasoning about the reasoning of other agents, which is not present here.

De Rosis and Carofiglio (deRosis, F; Carofiglio, V; Grassano & Castelfranchi, 2003) focus on the communicative per-

spective of a deceptive action. In their scenario, a Sender agent tries to convince a Receiver agent that some fact X is not true, where the Sender can lie or use other deceptive strategies. Their system, "Mouth of Truth" implements reasoning models as belief networks (Neapolitan, 1990; Pearl, 1997), where nodes represent belief and probabilities across links to other node represent uncertainty. This allows the Sender to lie not about the belief they want to manipulate, but about one connected to it. Thus uncertainty can be increased for the belief "it rained" if the Sender claims "the floor outside is dry". However, the Sender needs a model of the Receiver's beliefs to be able to do this and so acts as if its own set of beliefs and reasoning rules is replicated in the Receiver. This can then be used to influence the decision making process of the Sender.

The work so far discussed did not ground deception in an explicit model of other agents. Theory of Mind is a term coined by (Premack & Woodruff, 1978) who define it as the ability to infer the full range of epistemic mental states of others, i.e. beliefs, desires, intentions and knowledge. This is a mechanism that helps to make sense of the behaviour of others in specific contexts and to predict their next action.

Recent work (Harbers & Meyer, 2009) focuses on a computational implementation of ToM, giving agents the capacity to interact in a believable way with trainees, and to explain their actions and decisions after the training is over. The agents model a trainee's mind and give feedback either through simple action decisions, or by an explanation at the end. Meyer et al. here combined two prominent but conceptually different approaches to the human theory of mind: the Theory-Theory approach (TT) and the Simulation-Theory approach (ST).

In TT, the mental state we attribute to others is not observable, but is knowable through intuition and insight. Implementationally, this is achieved by using inference rules to reason about the beliefs of others. On the other hand, ST claims that each person simulates being another while trying to reason about their epistemic state, using the same structures and processes as those updating their own beliefs and knowledge (Aylett & Louchart, 2008). Meyer et al. showed that the main difference lay in ease of implementation rather than in outcome, as ST models are better in terms of code re-usability and modularity. Moreover, the TT approach can only deal with BDI (Beliefs Desires Intentions) models (Rao & Georgeff, 1995) due to a rigid representation of the mental state of other agents in terms of beliefs, limiting it to a specific symbolic representation.

PsychSim (Pynadath & Marsella, 2005) is a multi-agent based simulation tool for modeling interactions using a decision-theoretic approach. Unlike most such frameworks, where agents select actions maximizing rewards using their own beliefs, PsychSim agents also take into account their beliefs about other agent's beliefs. These recursively- "nested beliefs" may include subjective views of the agent itself. Agents update their beliefs according to the changes in the world and their subjective interpretations of world dynamics. In particular, messages are implicit ways through which one agent may influence the beliefs of another.

Wagner and Arkin developed algorithms to give an an intelligent robot the ability to deceive (Wagner & Arkin, 2010). The Deceiver seeks to induce a false belief in another agent, the Target, who is modeled as an action model and utility functions with associated outcomes matrix for a specific situation. This involves performing some action in the environment transmitting a false communication to the Target, so that it will behave in a way benefiting the Deceiver. This modifies the outcome matrix for the Target, the *induced outcome matrix*. Wagner and Arkin showed that knowledge of the Target affected the success of a deceit attempt. However this work did not explore the implications of different levels of ToM. Although there are systems that implemented a Theory of Mind in agents, and interesting projects on deception, we believe this is the first generic model that combines the two in a way that is flexible enough to be featured in a game. Further, we also show a study that compares different levels of abstraction in the way agents are perceived in terms of lying.

A Mindreading Agent Model

Our agent ToM is based on the Mindreading model of (Baron-Cohen, 1995), and follows the ST approach of Meyer et al.- see Figure 1. A central Knowledge Base (KB) stores the agent's beliefs and world knowledge and is the foundation for the agent's behaviour given that its actions are based on its knowledge.

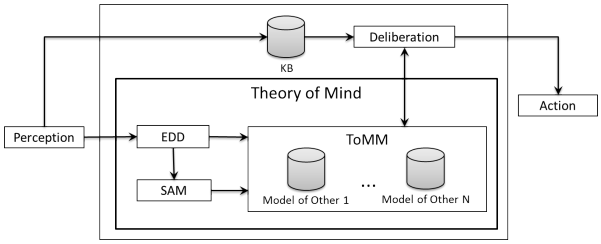

Figure 1: Proposed model for a Mindreading Agent

The ToM has three components, following Baron-Cohen[1]: the EDD (Eye Direction Detector), SAM (Shared Attention Model), and ToMM (Theory of Mind Mechanism). EDD determines who sees what, while SAM constructs higher level relations between entities (John sees that Luke sees the book). The ToMM represents and stores the mental states of other agents and is used to influence or deceive another agent. However, a deceiving agent must also be able to plan and reason about the consequences of its own actions. Thus our model includes a Deliberation component giving planning capabilities using knowledge from the KB and the ToMM to select the best actions for the agent to perform to meet its current goals.

[1]There is an additional component, the Intentionality Detector but to simplify our model it was not included

Representing Models of Others

Each Model of Other in the ToMM represents the beliefs of a specific Other the agent knows. A single-level theory of mind allows us to represent an agents's beliefs about another agent's beliefs. However, human adults are able to model more than one level (e.g. beliefs about another's beliefs about another's beliefs). Children start to develop a second level of ToM at around the age of six. Thus agents intended to function believably at the level of older children - as in the Werewolf game used as a study - require a model with more than one level of ToM.

A specific Model of Other contains its own ToMM also containing Models of Others, creating a recursive hierarchical tree-like structure - see Figure 2.

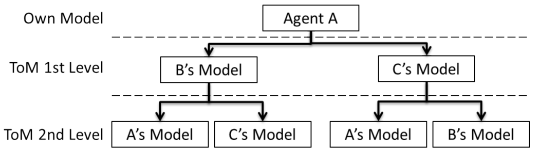

Figure 2: ToMM Hierarchy: 3 agents and 2 levels

Thus three agents, A, B, and C, each with a two-level ToM modeling ability, need six models each. If agents include a three-level ToM, this rises to fourteen models, and with four levels, to thirty models. The more complex the tree structure for the model hierarchy, the more effort is required for each update cycle. We will focus on a two-level ToM, bearing in mind that in the human case, applying more than two levels also causes a substantial overhead. More levels could be used in exchange for a slower reasoning cycle.

The ST approach represents others by simulating ones own processes in that same situation. Hence a ToMM Model of Other corresponds to a simplified version of the Agent Model depicted in Fig. 1, including both data structures and processes. A Model of Other can therefore be updated with a given percept through the same process used to update the agent's own model.

Updating Models of Others

When a given percept is received (e.g. a property has changed, or an action was performed), the agent updates its KB and its Models of Others. This is done through the EDD and SAM components.

The EDD determines what entities, objects, and events are perceived by other agents. It first checks whether a target agent is within a certain radius or in the same location as the agent, and if so, asserts that it also receives the percept.However this does not deal with more complex percepts such as a whisper into an ear, where only the specific receiving agent will know what was said. Hence the EDD may also include domain-specific rules about actions with particular restrictions on the perceptual mechanism. A rule specifies information about the action (such as subject, action name, target, parameters) and associates it with a list of effects. Two main types of effects are used in these rules:

- Global effect - effect of an action assumed to be perceived and shared by everyone (who is close enough). E.g. $*$:$Werewolf(Rob)$ represents that everyone can perceive $Werewolf(Rob)$.
- Local effect - an effect perceived only by a particular agent. E.g. $John$:$Werewolf(Rob)$ represents that only $John$ will perceive $Werewolf(Rob)$.

When EDD receives percept P, it determines two lists, perceptionVisibilities and agentVisibilities. The perceptionVisibilities list contains all pairs Ag:P, such that agent Ag perceives proposition P, while the agentsVisibilities list contains all pairs of form Ag:Ag, stating which agents see which other agents. SAM uses this to update Models of Others. It traverses the tree hierarchy, establishing whether a Model M should perceive P applying the following test:

1. Test if Model M is contained in *perceptionVisibilities*.
2. Test if the pair $Predecessor(M)$:M is contained in the agentVisibilities list. $Predecessor(M)$ returns the predecessor of model M in the tree hierarchy.
3. If both tests are verified, then model M can perceive P, otherwise the algorithm stops following the remaining subtree and continues the recursive process.

For example, suppose three agents, A, B and C. When A receives a percept P, it will update its own KB with P, but will also process P in its ToM to update models for B and C. Further, suppose that A knows that both B and C perceived P, and also knows that B does not see C (so it will not see that C perceives P). In this situation A's Model of B will be updated with P but A's model of B's Model of C (second level) will not be updated.

Using the ToMM Information

Agents have two reasoning mechanisms, one forwards (from data to conclusions) using inference rules, and one backwards (from goals to actions needed to achieve them) used to create plans that achieve the agent's goals. An inference rule is a tuple $< R,P,E >$ where R is the name of the rule, P (Preconditions) is a list of propositions that need to be verified for the rule to be applied, and E (Effects) a list of propositions that will be added to or removed from the KB when the rule is applied. Whenever new knowledge is added to the KB, the deliberation component will test the preconditions of the existing Inference rules. If any rule is fired (i.e. its preconditions are verified) the deliberation component will automatically update the KB with the effects in its effects list. If this process adds a new proposition to the KB, the inference process will be repeated until no more changes are verified.

The second mechanism involves goals, plans and actions. A goal is a tuple $< G,P,S >$ where G is the Goal's name, P a list of propositions that correspond to the goal's preconditions, and S a list of propositions that correspond to the goal's

success conditions (i.e. the desired goal state). The deliberation component is constantly checking to see if any goal becomes active by testing its preconditions. Once a goal becomes active, the planner tries to build a plan of actions to achieve the goal's success conditions. The actions used by the planner are defined using a STRIPS-like (Fikes & Nilsson, 1971) formalism and correspond to a tuple $< Ag, A, P, E >$ where Ag is the agent who performs the action, A is the action's name, while P and E correspond to a list of preconditions and effects. Given the similar representations, Inference Rules can also be used by the planner to build plans of actions; the difference is that when an Inference Rule is selected for execution (when the agent is executing the plan) it is not returned as an action to be performed in the environment. For more details about these mechanisms, please refer to (Aylett, Dias, & Paiva, 2006).

The first step in making the ToM information available to the deliberation component is to allow the specification of preconditions that are not tested against the agent's own KB but using a particular Model of Other. This is done by specifying explicitly the Model of Other to be tested by representing preconditions as a list of colon separated agents followed by a proposition Ag_1:...:Ag_n:P. When the deliberative component finds such a precondition it starts by traversing the tree hierarchy of Models of Others using the list of colon separated agents, and selecting the corresponding Model Of Other. Then the proposition P is tested using the selected Model of Other's KB. As example, A:B:$Suspects(A)$ is true if $Suspects(A)$ is true in the Model of B that is stored in the agent's Model of A (intuitively representing "'I think that A thinks that B suspects him to be the Werewolf"'). If a proposition P does not specify a Model of Other it will be tested against the agent's own model, in other words, its own KB.

Using preconditions this way allows us to specify goals and inference rules triggered according to beliefs of others. It would be even more useful to model higher-level goals and inference rules, i.e. explicit goals and rules to change the mental states of others. To do so, we use the same mechanism used to specify local and global effects as described previously. An effect is specified as Ag_1:...:Ag_n:P, where Ag_i is an an agent's name, or the symbol "'*"',' and represents that only the Models of Others obtained by the list Ag_1:...:Ag_n will have the proposition P added to its KB. The symbol "'*"' represents that all Models of Others at that particular level will be selected. The planner was extended to be able to handle matching and detection of conflicts between preconditions and local/global effects. In planning terms, a precondition is matched or threatened by a local effect only if their agents lists are compatible and if they refer to the same proposition P. In its simplest version, two agents lists are compatible if they have the same size and the agents are unifiable (the symbol "'*"' unifies with everything). As examples, the effect A:B:$Suspects(C)$ matches the precondition A:B:$Suspects(C)$, but does not match the precondition B:A:$Suspects(C)$, whilst A:$*$:$Suspects(C)$ matches both A:B:$Suspects(C)$ and A:D:$Suspects(C)$.

When an inference rule has an effect specified with an agents list (e.g Ag:P), instead of updating its own KB, the deliberation component will traverse the tree hierarchy in order to update the corresponding Models of Others. Moreover, the ST approach means that the Model of Other corresponds to a version of an Agent Model with its own inference mechanism. When creating a Model of Other, the agent assumes that others will use the same inference rules as its own. Therefore, every update cycle, the inference mechanism will also be executed recursively for each Model of Other. In other words, the agent will simulate other's inference processes, and update the corresponding models. This process is applied even if the effects of the inference rule specify an agent's list. For instance, if the Model of Other of John at level 1, applies an inference rule that results in the effect Rob:$Suspects(John)$, it will update John's Model about Rob's Model at level 2.

Due to its greater complexity, we did not include goal selection/planning, and thus simplified the version of the Agent used as a Model of Other. The agent is therefore not capable of simulating the planning process of others.

Case Study

The model above was used to build NPCs that deceive in a system for intercultural training, MIXER (Hall et al., 2011). This is aimed at children aged 9-11 and conflict between groups (an in- and out-group scenario) is presented through a social game. Rules act as cultural expectation and if they are varied, conflict will occur. Older children usually define rules before starting to play, but late primary children generally only discover the difference in rules when the conflict occurs, often with game abandonment and shouts of "'it's not fair"' and "'I don't want to play any more"'. The user acts as an invisible (out-of-game) friend to a character thrust into this situation with the pedagogic aim of showing that the existence of different rules is not the same thing as 'cheating'. MIXER uses variations of the game Werewolf, or Mafia[2].

A simplified version of the game involves five players, the Villagers, who are divided into two groups, one Werewolf and four potential Victims. Victims have limited information, since they do not know who the Werewolf is (they are 'killed' at night). Characters can be human players or NPCs (Non Playable Characters) running the architecture supporting deception. The goal is to discover who is the Werewolf: the character who is lying.. The Werewolf must lie purposefully: its objective is to remain hidden until no longer outnumbered by Victims. Thus it tries to eliminate Victims while concealing its true identity.

The game has been implemented in turn-based rounds. In each round every character performs the Accuse action in order, naming another character as the Werewolf (see Figure3). The Werewolf deceptively accuses one of the victims, knowing they are not in fact the Werewolf. At the end of each turn,

[2]http://en.wikipedia.org/wiki/Mafia_(party_game)

the agreed werewolf is excluded from the game and informs the other agents about its true identity. This is used to infer new information about past accusations. The real Werewolf wins if it reaches the last turn alive, when there is only one victim left. At this stage the Werewolf announces its identity. Victims win if they manage to discover who the Werewolf is before the last turn.

Figure 3: An agent performing the *Accuse* action

The following inference rules allow the victims to reason about past actions, trying to determine possible werewolf suspects:

- I suspect those that were accused by someone I don't suspect
- I stop suspecting someone who accuses a target I suspect
- I suspect those who accused a victim that was eliminated the previous round
- Someone who accuses a target suspects that they are a werewolf
- Someone that is accused will suspect the accuser

Modeling the Werewolf

Two versions of the Werewolf agent were implemented. One has a single-level ToM, able to represent what victims believe, but not what victims think it or the other victims believe. The second has a two-level ToM, able to represent what victims think about what it knows and in general, what victims think about the suspicions of others. Both versions also have the inference rules above, used by victims to determine suspects.

The single-level Werewolf has two main strategies compatible with its single-level ToM: eliminate victims that suspect it, and make a victim suspect another victim who has not been accused yet. The second goal corresponds to changing the victim's beliefs, and can be modeled by the success condition $[v_1]$:*Suspects*$([v_2])$, where $[v_1]$ is a variable representing a victim and $[v_2]$ is a variable representing another victim. These variables will be instantiated by the goal activation process, and the agent will then try to make $[v_1]$ suspect $[v_2]$.

The two-level Werewolf agent has a strategy commonly used by human players in this game. The agent will "'Lay low"', by avoiding suspicious actions, trying to make victims believe that it thinks the same way they do. This is modeled with the following second level success condition $[v]$:*SELF*:*Suspects*$([target])$, where $[v]$ is a victim, $[target]$ is another villager that $[v]$ suspects to be the Werewolf, and *SELF* represents the Werewolf agent itself. Thus the two-level ToM Werewolf will accuse villagers that are already being accused by other victims.

Tests and Evaluation

Two tests were run comparing these two versions. A first simulation test assessed how well the two types of ToMs performed in the game. In order to test the hypothesis that an agent with a single-level ToM is less successful in believable deception then an agent with a two-level ToM, a second evaluation was conducted with users, assessing their perception of the single-level and two-level ToM Werewolves.

As an autonomous agent architecture is being used, scenarios are unscripted and do not run identically. To avoid different outcomes biasing user responses, a video of a particular run was used for the second test. The simulation test allowed us to select this video.

In the first test, two versions of the system were generated. The first was parameterized so that the Werewolf used the single-level ToM (ToM1 condition), and in the second it used the two-level ToM (ToM2 condition). The victims used a single level ToM in both conditions. With five players, the maximum number of possible rounds is 4. Both versions ran ten times, from the beginning until the Werewolf was caught or won the game. The number of turns the Werewolf managed in each run was recorded. The video of the best scoring run for each version was used for the second test.

In the TOM1 condition, the Werewolf never lasted four rounds, and so did not win a single game. The ToM2 Werewolf won in two out of ten runs and on average lasted 0.6 more turns than the ToM1 version.

User Perception of the Lying Agents

The second test evaluated user-perceptions of the two Werewolf versions' believability. An online questionnaire was used with the two videos selected from the first test. Sixty participants (34 M, 26 F), of which 55 were aged 19-25, were recruited online, and randomly assigned to one of the two versions. They were asked to pay special attention to agents' actions and to try to work out who was lying. They watched the game and then rated affirmations using a Likert scale (ranged from -2 meaning totally disagree, to 2 meaning totally agree) in four sections: (1) affirmations about the game itself; (2) affirmations about all players; (3) the same affirmations as (2) but only for the the liar; (4) affirmations focused on deceptive behaviour. Data was analyzed using a non-parametric Mann-Whitney statistical test to compare conditions ToM1 and ToM2.

Participants perceived the ToM2 condition as more interesting according to *A1: "'The game is interesting"'* ($p < 0.05$, $r = -0.263$) and would play this version of the game more *"'A2: I would play a game like this"'*($p < 0.05$, $r = -0.292$). ToM2 scores were significantly lower ($p < 0.05$) for *A3: "'It is easy to win while playing as a Victim"'* and significantly higher ($p < 0.5$) for *A4: "'It is easy to win while playing as a Werewolf"'*. We conclude that participants

thought the liar did a more competent job in the ToM2 version.

Answers to *A8: "'Players behaved in a predictable way"'* were also significantly different in the ToM2 condition ($p < 0.001$, $|r| = 0.5$): player characters were seen as less predictable in ToM1 than ToM2. This is seen as a surrogate for believability given the answers to *A10: "'Players are easily deceived"'* gave significantly lower values in ToM1 than in ToM2 ($p < 0.001$, $r = -0.478$), reflecting the more believable performance of the Werewolf in ToM2.

Finally, two additional measures: "how well did the liar play" and its "intelligence" also lead to statistically significant differences between the two conditions ($p < 0.001$) supported by a large effect size($|r| = 0.5$). We conclude that the liar in ToM2 is perceived as more intelligent than in ToM1. Also statistically significant ($p < 0.001$, $r = -0.467$) were answers to *A15: The liar is affected by others' actions* indicating that the Werewolf was seen as more responsive to the play of others in ToM2. Finally, the higher results in ToM2 for *A21: The liar managed to deceive the other players* ($p < 0.001$, $r = -0.524$) confirm those for A10 above.

Conclusions

This paper advances a model for virtual agents that are able to deceive, embedding a ToM mechanism inspired by work on the human ToM. The model can produce N-level ToM behaviour using a simulation approach, where the agent runs its own mechanisms, reasoning about the beliefs and actions of others as if it was in their shoes. Parametrization allows the number of levels of ToM to be easily varied.

Evaluation was carried out using a social game, MIXER, for intercultural training of children aged 9-11. This game includes one character, the Werewolf, that must lie in order to play successfully. The first test showed that when a Werewolf was given a two-level rather than single-level ToM and played against Villagers with a single-level ToM, the Werewolf's game performance improved. The user testing with 60 subjects showed that participants clearly perceived the ToM2 version Werewolf as better at deceiving the other agents, and, furthermore, saw this as more intelligent behaviour. These results support the hypothesis that an agent with a single-level ToM will be less successful in believable deception then an agent with a two-level ToM.

As future work, it would be interesting to compare different combinations of the scenarios (e.g. one-level werewolf against two-level victims), and to include the simulation of other's planning processes in order to make it posible to reason about other agent's goals and plans. Another interesting extension to this work would be to apply the model to a different type of deception than verbal lies, for example to deceptive display of affective states (Rosis et al., 2003).

Acknowledgments

This work was partially supported by the European Community (EC), through the ECUTE project (ICT-5-4.2 257666), and by national funds through Fundação para a Ciência e a Tecnologia (FCT), under project PEst-OE/EEI/LA0021/2011. The authors are solely responsible for the content of this publication. It does not represent the opinion of the EC or the FCT, which are not responsible for any use that might be made of data appearing therein.

References

Aylett, R., Dias, J., & Paiva, A. (2006). An affectively-driven planner for synthetic characters. In *Proc. icaps.*

Aylett, R., & Louchart, S. (2008). If I were you: double appraisal in affective agents. In *Aamas-volume 3* (pp. 1233–1236).

Baron-Cohen, S. (1995). *Mindblindness: An Essay on Autism and Theory of Mind*. MIT Press. Paperback.

Bates, J. (1994). The role of emotion in believable agents. *Communications of the ACM*, *37*, 122–125.

Castelfranchi, R., C; Falcone, & deRosis, F. (1998). Deceiving in Golem: How to strategically pilfer help. In *Autonomous agents: Workshop on deception, fraud and trust in agent societies.* Kluwer.

deRosis, F; Carofiglio, V; Grassano, G., & Castelfranchi, C. (2003). Can computers deliberately deceive? a simulation tool and its application to Turing's imitation game. *Computational Intelligence*, *19*(3), 235-263.

Dias, J. a., & Paiva, A. (2005). Feeling and reasoning: A computational model for emotional characters. In C. Bento, A. Cardoso, & G. Dias (Eds.), *Progress in artificial intelligence* (Vol. 3808, p. 127-140). Springer.

Fikes, R. E., & Nilsson, N. J. (1971). Strips: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, *2*(3-4), 189-208.

Hall, L., Lutfi, S., Nazir, A., Hodgson, J., Hall, M., Ritter, C., et al. (2011). Games based learning for exploring cultural conflict. In *Aisb 2011 symposium: Ai & games, york* (pp. 6–7).

Harbers, K. v. d., Maaike; Bosch, & Meyer, J.-J. (2009). Modeling agents with a theory of mind. In *Proc. 2009 ieee/wic/acm int. joint conf. web intelligence and intelligent agent technology - v. 02* (pp. 217–224). Washington, DC, USA: IEEE.

Marsella, S., & Gratch, J. (2009). Ema: A process model of appraisal dynamics. *Cognitive Systems Research*, *10*(1), 70–90.

Mateas, M., & Stern, A. (2003). Façade: An experiment in building a fully-realized interactive drama. In *Game developers conference, game design track* (Vol. 2, p. 82).

Neapolitan, R. E. (1990). *Probabilistic reasoning in expert systems: theory and algorithms*. New York, NY, USA: John Wiley & Sons, Inc.

Paiva, A., Dias, J., Sobral, D., Aylett, R., Woods, S., Zoll, C., et al. (2004, July). Caring for agents and agents that care: Building empathic relations with synthetic agents. In *Aamas'2004*. ACM Press.

Pearl, J. (1997). *Probabilistic reasoning in intelligent systems : networks of plausible inference*. Morgan Kaufmann. Paperback.

Premack, D., & Woodruff, G. (1978). Does the chimpanzee have a theory of mind? *Behavioral and Brain Sciences*, *1*(04), 515–526.

Prendinger, H., & Ishizuka, M. (2001). Social role awareness in animated agents. In *Autonomous agents* (pp. 270–277).

Pynadath, D., & Marsella, S. (2005). Psychsim: Modelling theory of mind with decision-theoretic agents. In *Ijcai* (p. 1181-1186).

Rao, A. S., & Georgeff, M. P. (1995). Bdi agents: From theory to practice. In *Proc 1st int. conf. multiagent systems.* San Francisco.

Rosis, F., Pelachaud, C., Poggi, I., Carofiglio, V., & Carolis, B. (2003). From Greta's mind to her face: modelling the dynamics of affective states in a conversational embodied agent. *Int. J. Human-Computer Studies*, *59*(1), 81–118.

Searle, J. (1976). A classification of illocutionary acts. *Language in Society*, *5*, 1–23.

Swartout, W., Gratch, J., Hill Jr, R., Hovy, E., Marsella, S., Rickel, J., et al. (2006). Toward virtual humans. *AI Magazine*, *27*(2), 96.

Wagner, A., & Arkin, R. (2010). Acting Deceptively: Providing Robots with the Capacity forDeception. *Int. J. Social Robotics*, 1-22–22.

Wyer, B. D. D. K. S. K. M., & Epstein, J. (1996). Lying in everyday life. *Journal of Personality and Social Psychology*, *70*, 979-995.