Social Situated Agents in Virtual, Real and Mixed Reality Environments

M. Dragone[1], T. Holz[1], B.R. Duffy[2], G.M.P. O'Hare[1]

[1]Department of Computer Science,
University College Dublin, Belfield, Dublin 4, Ireland
{Mauro.Dragone, Thomas.Holz, Gregory.OHare}@ucd.ie
http://prism.cs.ucd.ie

[2]Institut Eurécom
2229 Route des Crêtes, BP 193 - F 06904, Sophia-Antipolis, France
Brian.Duffy@eurecom.fr

Abstract. This paper details a framework for explicit deliberative control of socially and physically situated agents in virtual, real and mixed reality environments. The objective is to blur the traditional boundaries between the real and the virtual and provide a standardized methodology for intelligent agent control specifically designed for social interaction. The architecture presented in this paper embraces the fusion between deliberative social reasoning mechanisms and explicit tangible behavioural mechanisms for human-agent social interaction.

1. Introduction

To date, research in intelligent virtual agents can be generally placed along a spectrum with two differing perspectives [1]: research focusing on the physical aspects of the virtual agent, where the aim is to try to reproduce the physical attributes of natural agents (such as modelling artificial fish [2] or virtual humans [3]); and research focusing on deliberation, user modelling and, in general, more abstract high level capabilities. Such classification effectively draws an arguable distinction between mind, body and behavioural context. This work, in addition, blurs the boundaries in a third direction, between the real and the virtual (often viewed as delineated) and aims to facilitate the integration of situated virtual and real agents in social deliberative interaction with humans.

In developing sophisticated control paradigms, robotics research has also provided a rich arena for intelligent reasoning systems as applied to real world contexts, with the field of intelligent agent research providing numerous strategies. While an obvious synergy exists between the two often viewed as disparate domains, few have strongly embraced the inherent advantages of achieving a coherent synthesis between the fields of intelligent agents, virtual characters, and intelligent robot control paradigms.

In order to develop a coherent framework for socially situated agents in multi-reality environments, this work draws on previous research in the field of autonomous social robotics, an arena where considerable research has been undertaken in recent years in developing the social deliberative capabilities of artificial systems [4,5]. One of the core methodologies employed in this paper is that of behaviour-based synthesis between perception-acting and deliberation as found in recent robotic research. Instrumental in the development of mobile agent technologies and cross-reality migration, is the work undertaken by the Agent Chameleons project [6].

In order to situate this work within the current state of the art, section 2 briefly discusses relevant control strategies as applied to virtual and real agents. This sets the stage for the Social Situated Agent Architecture (SoSAA) introduced in section 3.

2. Related work

Over the years, different control strategies for virtual agents have been proposed and implemented. Isla et al. [7], for example, propose a layered model for an artificial brain, where different layers communicate via a shared blackboard, allowing high-level functions to control lower ones (*subsumption*, cf. [8]) and vice versa (*supersumption*). They distinguish between sensing (noticing a stimulus) and perceiving (assigning meaning to a stimulus), allowing different perceptors to extract meaning from the same sensor. The agent's action selection mechanism is governed by a function that looks for the highest expected reward among the possible actions. Egges et al. [9] employ Finite State Machines to control the behaviour of a virtual, conversational agent that takes into account the perceived emotion of the user (via face recognition techniques) and the personality and emotional state of the agent. Chittaro and Serra [10] use a similar approach in the decision process of their agents, applying personality factors to Finite State Machines, but the influence is modelled probabilistically to further the realism of the agent by making it less predictable.

Although some of these systems use personality and emotion to promote agent believability, they are generally based on *reactive behaviour*, i.e. directly mapping perception to action. Cognitive agents, on the other hand, are inspired by models of human-like cognition, allowing the agent to deliberate about, and reflect upon these perceptions and actions before taking an action. De Rosis et al. [11], for example, use Dynamic Belief Networks [12] to model the mind of their believable conversational agent Greta.

One of the most popular and most widely researched cognitive models is that of BDI (Belief, Desire, Intention) agents [13]. BDI theory has proven a particularly apt methodology for autonomous agents in modelling human practical reasoning and grounding traditional symbolic reasoning in situations requiring real-time reactivity.

This work adopts the stance that the future lies in the central area of the spectrum between reactive agents and cognitive agents, where a fusion of the two is necessary. The framework and its implementation presented in the following sections aims to achieve a coherent synthesis between grounded perception-acting and BDI agent-based deliberation.

3. Social Situated Agent Architecture (SoSAA)

The Social Situated Agent Architecture (SoSAA) is a design methodology originally emerging from ongoing research with autonomous social robotic systems [14,4]. The SoSAA seeks to develop autonomous, rational, resource bounded, social and intentional agents, which can demonstrate an ability to perceive their environment, deliberate about their future and directed actions, and opportunistically form collaborative alliances with other agents (robots or humans) situated within their multi-reality environment. In investigating numerous control strategies capable of dealing with time and resource constraints, and uncertain and partial perceptions in typically noisy and dynamic environments, this work has embraced the synthesis between reactive and deliberative methodologies in order to achieve a coherent integration of representational and non-representational approaches.

The SoSAA can be conceptually decomposed into a number of fundamental levels; reactive, deliberative, and social, as outlined in the following sections. It is important to note that there is a strong interplay between these levels in order to achieve a structured integration of the system's functionality and its subsequent robustness.

3.1 Reactive-Behavioural level

As in [15], in designing the SoSAA reactive level, a *divide-and-conquer* strategy was adopted, breaking down complex actions into primitive control units called behaviours. Each behavioural unit performs a mapping between sensorial inputs, internal states and a robot's actions in an attempt to accomplish a specific goal (i.e. keeping a constant distance to the wall).

The SoSAA includes a behavioural suite which is the result of the ongoing effort in identifying a set of navigational and behavioural primitives for autonomous mobile robots. These primitives implement both reflex robot responses to unexpected or dangerous events (i.e. *stop on collision*) and more complex actions (i.e. *follow wall*, *move toward goal*).

The reactive level functionality is organized into a reactive controller component, which is responsible for the management of every activity (i.e. sensor drivers) and aforementioned behaviour functions (for a more accurate description see [14]. The reactive controller performs a tight closed loop between sensing and acting. At each cycle, the sensor's outputs are routed to the set of active behaviours and the resulting commands redirected to the relevant effectors.

Some behavioural systems (i.e. the Fuzzy Control of the Saphira Architecture [15] implement blending mechanisms that merge behaviour outputs in order to handle more complex situations while still relying on simple behavioural modules. This work argues that there are few cases that justify supporting behavioural blending in general.

In contrast, this work instead on a specific assemblage of behaviours obtained through traditional object-oriented methodology and the possibility of having more than one behaviour active at any given time covering different effectors or devices (i.e. arm grip, wheels). The navigational capabilities of the robots used, for instance, are based upon seminal methods for real time mobile robot obstacle avoidance like the Vector Field Histogram Plus [VFH+] [16] and the Dynamic Window algorithm

[17]. The basic obstacle avoidance behaviours consider the disposition of the obstacle in the vicinity of the robot – found, for example, by examining the output of the 2-D range-finder – to deduce a *set of feasible directions.* These are obtained by examining all the manoeuvres available to the robotic platform and excluding those leading to a collision within a pre-determined timeframe. The set of feasible directions may then be used to trade between different objective components. For example, each direction can be evaluated in relation to different aspects like a measure of the control effort (i.e. the acceleration required), the position of reference targets (i.e. for way-point navigation), or the distance from obstacles. The resulting manoeuvre is finally selected by maximizing a weighted sum of these evaluations. By balancing the weights of the components in different ways, different behaviours emerge. An example of this can be seen in section 4.2.

3.2 Behavioural – Cognitive Synthesis

BDI reasoning is based upon mental attitudes representing the informational (*beliefs*), motivational (*desires* and *goals*), and deliberative (*commitments*) states of the agents. These attributes provide the agent with a usable description of the present and future states of the agent's environment. This description may not necessarily be a faithful representation of the true state of the system, nor of the consequences of the agent's actions, as it would normally be expected of a traditional logic planning systems. A BDI agent's belief is instead a subjective statement of what the agent believes to be true at the current moment, with regard to its own state, the state of the environment, or the state of other agents in its environs.

Consequently, in order to account for incomplete and incorrect information, BDI agents generally employ temporal epistemic logic to deliberate upon their beliefs and find a suitable agent conduct. The BDI methodology decomposes the latter problem into primarily two stages. Firstly, certain facts are included in a set of agent desires (the statements representing states that the agent wishes to be true); secondly, suitable courses of actions are identified as a set of commitments of the agent (each commitment representing a state that the agent is committed to achieve). The second stage usually takes the form of means-end reasoning mechanisms.

SoSAA adopts a constructional approach to bridge the gap between BDI theory and practice (see [18]). In this work, the practical logic reasoner and planner is delivered through Agent Factory [19], an integrated environment for the rapid prototyping of social intentional agents. This system, while simplifying certain aspects of the BDI methodology, provides clear constraints on the agent computational model through the definition of the strategies controlling, for example, the selection of goals or the reconsideration of commitments. SoSAA complements the architectural constraints embedded in Agent Factory with a number of design tools [20] and guidelines, which facilitates the design of BDI style agents and their instantiation in a number of different domains.

Core to the architecture is the Object Tracking subsystem. This subsystem implements an anchoring mechanism, which is similar to the *Artefacts* in the Saphira architecture or to *Sensorial Anchoring* in [21]. The subsystem creates and maintains the connection between symbols and physical objects over time (even if they temporarily disappear from the field of view), identified through the robot's sensorial apparatus. The subsystem also manages to notify the cognitive layer of meaningful events in conjunction with significant changes in the state of the perceptual space of the robot (i.e. start_tracking(object), close(object)).

A soccer player robot, for instance, will be able to reason about objects not directly sensed, without attempting inappropriate activities such as kicking when not in control of the ball, or avoidance of nonexistent objects or, even worse, cancelling pursuit of the ball when it becomes occluded.

A key issue of the interface between the behavioural and the cognitive layer is the interplay between reactive and cognitive control. The deliberation process should not be inundated with requests to deduce new facts and commitments based on every numeric change in the reactive layer (i.e. the position of a tracked object). The agent instead should be able to describe – based on the context of the current task - meaningful geometric relationships between objects to which it intends to respond. For this purpose, the Object Tracking subsystem extracts basic qualitative representation of the situation surrounding the robot. The mechanism is based upon the *Constraints Ontology for Qualitative Reasoning* [22]. In it simplest form, the value space for the variables residing inside behavioural modules is partitioned, defining meaningful landmark values, and subsequently used to create qualitative representations. In addition to the interplay problem, with an increasing number of events computational issues may arise. SoSAA addresses these issues with functional partitioning of the reasoning process. The sensor information at the physical level, for example, is abstracted and organized into intermediate representations following a hierarchical organisation based upon increasing levels of persistence. As in [23] these intermediate representations form the basis of partitioning the deliberative process, defining regions of competences and dependencies among functional areas.

Consequently, the SoSAA cognitive level follows a Multi-Agent-System (MAS) organization with several agents supervising the different functional levels of the robot. At any given time, a number of agents share the control of the robotic platform. These agents vary in complexity from simple procedural knowledge modules that deal with lower level capabilities of the platform (i.e. sensorial organization, configuration and behavioural sequencing) to means-end reasoning (i.e. path-planning).

An important domain-specific issue for autonomous agents sensing and acting in the real world is the creation of beliefs from uncertain and noisy information. The SoSAA Behavioural Level incorporates perception units in association with its behavioural modes. In observing that sensory-motor primitive constrains the dynamic of the interactions between the robot and its environment, this constitutes an effective motivation to perception structuring and attention focusing. In earlier work [24], it has been shown how behavioural modes simplify the perceptual space and how feature detection (i.e. identifying signatures in the values returned from the sonar ring during wall following) can be used to create perception hypothesis and expectations in order to channel future structured sensing strategies, leading to the formation of perceptual evidence.

3.3 Social Intentional Agents

A distinguishing feature of the Agent Factory-developed deliberative level of SoSAA is its support for explicit social interaction in the form of a social level implanted in each of its agents. This social level is charged with maintaining a model for every agent acquaintance so that their behaviour can be accounted and influence the reasoning process. To facilitate collaboration among agents, Agent Factory agents make use of Speech Act Theory [25], a formalism for accurate and expressive communication mechanisms in Multi-Agent Systems. This is undertaken by performing a speech act (such as *requesting, ordering, informing* or *promising*) that sends a message to one or more of their socially capable acquaintances in order to affect their mental states. In this work, the robotic agents interact via Agent Communication Language (ACL) directives with semantic corresponding to the specifications outlined within FIPA (Foundations of Intelligent Physical Agents, see http://www.fipa.org). At a simple level, the messages received may trigger specific commitment rules governing the reaction of the receiving agent. The following example (in pseudocode) illustrates how a robot playing soccer, when asked to move to its home position (*`reset`*) on the football pitch, adopts the appropriate commitment.

```
BELIEF(requested_achieve(reset) & BELIEF(role(?R)) &
BELIEF(Home (?R,?X,?Y)
=> Commit(Self, Now,
ActivateBehaviour(MoveTo(x,?X,y,?Y)))
```

In addition to FIPA “`inform`” and “`request`” directives, a number of more sophisticated interaction protocols have also been implemented, among them, the Contract-Net-Protocol, which is used in group formation or task allocation for example.

4. The Social Situated Agent Architecture in Action

The Social Situated Agent Architecture provides for multi-reality implementations. As the SoSAA employs embodiment abstraction strategies implemented across its multi-layered architecture, it facilitates instantiations within virtual, physical and mixed reality environments. At the cognitive layer, SoSAA makes use of the embodiment mechanism of Agent Factory. This defines `Actuator` and `Perceptor` modules for interfacing to diverse applicative domains and provides a framework for reasoning about embodiment forms in terms of agent capabilities and constraints [14]. A degree of abstraction from the sensor and actuator modalities is also achieved in the reactive-behavioural layer where the physical level is individually tailored to each hardware platform. `Behaviour` implementations do not address the specifics of what body they are controlling, thus enabling easy portability of code from simulated to physical robots of differing platforms. The following examples illustrate how the system has been instantiated with a view to demonstrating the systems flexibility and versatility.

4.1 Physical Agents

Figure 2 illustrates a section of the specifications for a single robotic agent (a Nomad Scout robot) fetching a coloured ball and bringing it to its home position.

SoSAA Agent specifications are stored in ASCII files containing Agent Factory Agent Programming Language (AF-APL [18]) scripts. AF-APL scripts contain initial beliefs; the declaration of actuators and perceptors in use by the robotic agent and commitment rules governing behavioural transitions, plan activation, and goal decomposition. A Platform Manager Agent constitutes the main script, which describes the robotic agent and supervises its initialisation. This script can also contain a list of references to additional AF-APL scripts (i.e. roles and plans), each specifying the BDI design for a different functional area.

Figure 3 shows key snapshots from the execution of the *fetch ball* task. The robot can be seen approaching the ball using its estimated coordinates - as deduced by the camera activity that performs colour-segmentation on the image captured from the on-board camera - as way-point targets for its obstacle avoidance behaviour. Thereafter, when the ball is judged sufficiently close, a PID (Proportional Integrative Derivative) controller is selected as the behaviour of choice to control the gaze of the robot and direct the acquisition of the ball. Once the robot is in control of the ball, it turns and returns to its home position, reactivating the obstacle avoidance behaviour.

The DEFINE macros in the first part of the script describe simple landmark values for a qualitative description (*close/distant/touching*) of the distance of the ball.

```
ACTUATOR PRISM.RobotAgent.SetupActuator
ACTUATOR PRISM.RobotAgent.ActivateBehaviourActuator
PERCEPTOR PRISM.RobotAgent.ActionPerceptor
PERCEPTOR PRISM.RobotAgent.EventsPerceptor
…
DEFINE close(Ball) RobotCtrl.Tracking.ObjectTrracked.distance < 1000    // close if less than 1m
DEFINE distant(Ball) RobotCtrl.Tracking.ObjectTracked.distance >= 100  //  distant otherwise
DEFINE touching(Ball) RobotCtrl.Tracking.ObjectTracked.distance < 50   // touching if closer than 5cm
…
BELIEF(start) ⇒ COMMIT(Self, Now, ActivateBehaviour(Stop))
BELIEF(start_tracking(ball)) & (BELIEF(distant(ball))
          ⇒ COMMIT(Self, Now, ActivateBehaviour(MoveTo(Object,ball,MaxV,100))
BELIEF(end_tracking(ball)) ⇒ COMMIT(Self, Now, ActivateBehaviour(Scan, timeout, 5000)))
BELIEF(timeout_Scan) ⇒ COMMIT(Self, Now, ActivateBehaviour(MoveFree, timeout, 20000)))
BELIEF(timeout_MoveFree) ⇒ COMMIT(Self, Now, ActivateBehaviour(Scan, timeout, 5000)))
BELIEF(close(ball)) & BELIEF(sensing(ball)) &
!BELIEF(current(FaceObject)) &  !BELIEF(touching(ball))
        ⇒ COMMIT(Self, Now, ActivateBehaviour (FaceObject (Object
         ,ball,MaxV,40,w,450,aw,300,PID,1000,0.2,0)))
BELIEF(start_touching(ball))
        ⇒ COMMIT(Self, Now, ActivateBehaviour(TurnToward(X,0,Y,0,MaxV,200)))
BELIEF(turned) ⇒ COMMIT(Self, Now, ActivateBehaviour(MoveTo(X,0,Y,0,MaxV,100)))
```

Fig. 1. AF-APL Script controlling the fetch-ball task

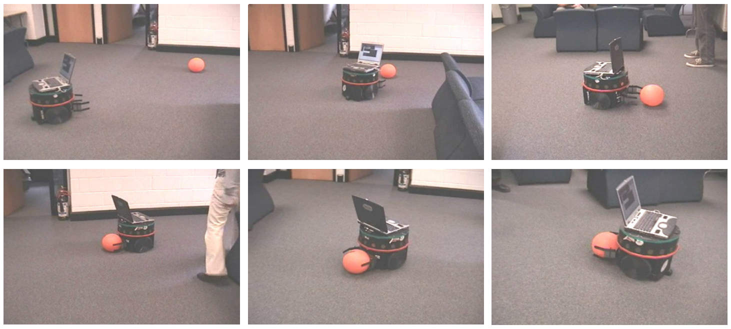

Fig. 2. A Nomad Scout robot fetching a coloured ball

4.2 Virtual Agents

Using Virtual Environments for simulation, prototyping, and testing of robotic control architectures is an obvious and widely employed approach, as experimenting in the real world can prove both, complicated and costly. SoSSA comprises of a set of simulated sensors and effectors interchangeable with the real world counterparts. Figure 4 shows a simulated robot performing the fetch-ball task in a virtual space. The simulated robot is under control of the same AF-APL script as the real robot (see Figure 2). The only difference is that all sensor drivers and actuators have been replaced with simulated objects. The emphasis in this work is on the faithful replication of real behaviours. By mirroring simple behaviours in virtual space (i.e. emulating noises and timing of the sensorial apparatus), all layers of the SoSSA architecture can be subsequently exercised.

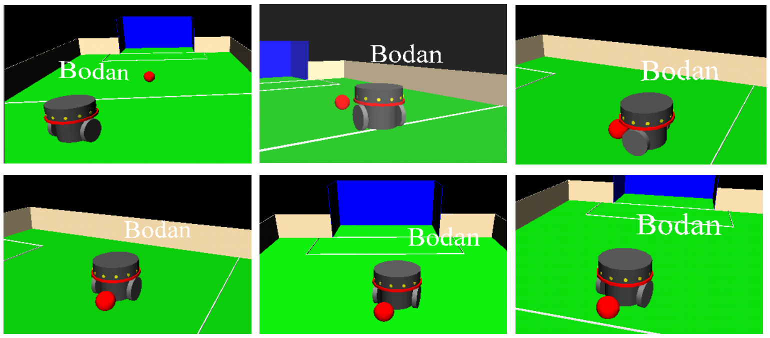

Fig. 3. A simulated Scout robot fetching a coloured ball

While the degree of complexities existing in real world environments is not found in artificial virtual spaces, there are advantages in transferring robotic architectures to virtual agents. Real-world robotic architectures are usually more robust, as they have to cope with a more complex, a more unpredictable, and a more uncertain world.

The BDI methodology is particularly well suited for the creation of believable characters as its cognitive framework facilitates the implementation of subjective behaviours.

Figure 5 shows three different views of a virtual environment populated by a group of virtual robots and other artificial characters animated using AF-APL scripts. In the example each agent is under the control of a different behaviour obtained with different weights of the components in the SoSAA obstacle avoidance module.

Figure 5 (left) shows the different trajectories followed by each agent. The set of weights for the behaviour of the robot “Bodan” (in the corridor) are set to maximize the speed of the robot. The agent “Bunny” is instead performing the *wall following* behaviour, favouring manoeuvres that approach the closer obstacle on the left of the robot. The robot “Bui” (in the right-lower room) is using an *Escape* behaviour which brings it to prefer to stay clear of obstacles. Finally, the agent “Snowman” is static, permanently located in the corridor. Its script controls an animation effector which makes him salute the user (by waving its hat) when the avatar of the user gets in its proximity. The right picture exposes the perceptual state of the robot “Bodan”, showing in the example the detection of a door and other objects through its range and vision sensors.

The virtual world in the example has been implemented with the Virtual Robotic Workbench [26], our Multimedia Collaborative Virtual Environment framework for communities of intentional agents.

Fig. 4. Views from a Test Virtual Environment. Left: Agent trajectories. Middle: User view. Right: Perspective and tracked objects from Robot Bodan

4.3 Mixed Agents

Having previously considered our physical and virtual agent cousins we now consider how SoSAA can accommodate a hybrid of these capabilities within an Augmented Reality scenario. In order to correctly align the virtual images with the real scene the user’s position and orientation has to be tracked. An efficient and cost-effective way to do this is ARToolkit [27], a system that facilitates the recognition and pose estimation of physical markers. We arranged five markers in a cube (Figure 6 (a)) to make the robot traceable from all angles. The SoSAA makes the user’s point of view known to the robot, which then turns to the user and greets him via its virtual avatar. The agent thus makes a combined use of its physical and virtual embodiment.

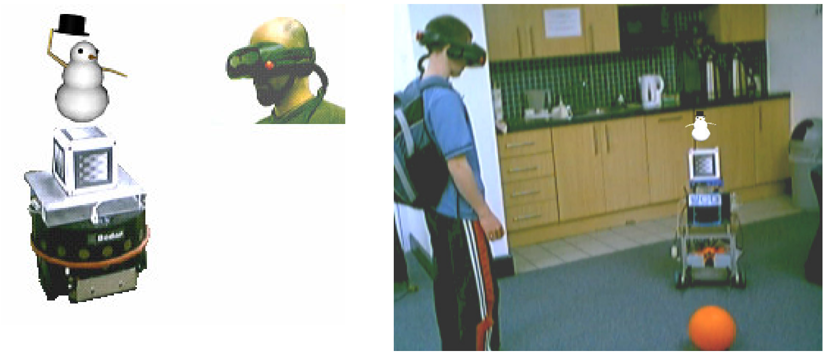

Fig. 5. (a) The agent turns to the user to greet him. (b) The agent fetches the ball for the user

Figure 6 (b) shows a snapshot of the robot fetching the ball when the user requests it. Since the robot knows the position of the observer, it can bring the ball to the user's location.

This work incorporates strong notions of perceptual identity in artificial systems through the use of stereotypes, character (perceived identity) and roles [26]. The SoSAA framework provides a flexible mechanism where users can customise both the agent's virtual persona and how this is managed through explicit mechanisms for artificial identity. While each agent's representation is fundamentally grounded on a *unique* identity, these personalisation mechanisms allow users to select their own preferred avatars in both virtual and augmented reality applications. In such a Mixed Reality environment, the SoSAA supports not only different users seeing different avatars, but also facilitates users with no equipment such as a Head Mounted Display (HMD). Such participants would therefore only see the robot's physical body and as such would only interact at the physical level.

5. Conclusions

The primary objective of the work presented in this paper has been to introduce a framework for explicit social interaction between people and a situated deliberative agent. This agent can manifest itself through a virtual avatar or an augmented reality agent in conjunction with a physical robot. The concept of artificial identity is specifically addressed to augment persistent social grounding between people and artificial systems. The result is a flexible infrastructure which allows for the rapid prototyping of social situated agents.

Numerous different implementations of the SoSAA have been undertaken which clearly fuses the notion that a physical robot is in fact a physically embodied agent. The system's context and environmental situatedness simply provides a different data set for deliberation and reactive behaviour. While it is argued that physical embodiment is a necessary criterion for the development of artificial intelligence, this work adopts the stance that an inherently artificial system is fundamentally constrained by its artificiality and as such can exploit quite different frames of reference.

Acknowledgments

This research is based upon works supported by the Science Foundation Ireland under Grant No. 03/IN.3/1361.

References

[1] Aylett, R. and Luck, M.: Applying Artificial Intelligence to Virtual Reality: Intelligent Virtual Environments. Applied Artificial Intelligence 14 (2000) 3-32

[2] Terzopoulos, D., Tu, X. and Grzeszczuk, R.: Artificial Fishes: Autonomous Locomotion, Perception, Behavior, and Learning in a Simulated Physical World. Artificial Life 1 (1994) 327-351

[3] Thalmann, D.: Challenges for the Research in Virtual Humans. In: Proceedings of the Workshop on Achieving Human-Like Behavior in Interactive Animated Agents, Barcelona, Spain (2000)

[4] Duffy, B.R.: The Social Robot. Ph.D. Thesis, University College Dublin, Department of Computer Science (2000)

[5] Dragone, M.: The Virtual Robotic Workbench. Technical Report No. UCD-PRISM-05-11. University College Dublin, Department of Computer Science (2005)

[6] Duffy, B.R., O'Hare, G.M.P., Martin, A.N., Bradley, J.F. and Schön, B.: Agent Chameleons: Agent Minds and Bodies. In: Proceedings of the 16th International Conference on Computer Animation and Social Agents (CASA '03), New-Brunswick, USA (2003)

[7] Isla, D.A., Burke, R.C., Downie, M. and Blumberg, B.: A Layered Brain Architecture for Synthetic Creatures. In: Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI '01), Seattle, USA (2001)

[8] Brooks, R.A.: Intelligence without Reason. In: Proceedings of the 12th International Joint Conference on Artificial Intelligence (IJCAI '91), Sydney, Australia (1991)

[9] Egges, A., Kshirsagar, S., Zhang, X. and Magnenat-Thalmann, N.: Emotional Communication with Virtual Humans. In: Proceedings of the 9th International Multmedia Modeling Conference (MMM '03), Taiwan (2003)

[10] Chittaro, L. and Serra, M.: Behavioural programming of autonomous characters based on probabilistic automata and personality. Computer Animation and Virtual Worlds 15 (2004) 319-326

[11] de Rosis, F., Pelachaud, C., Poggi, I., Carofiglio, V. and De Carolis, B.: From Greta's mind to her face: modelling the dynamics of affective states in a conversational embodied agent. International Journal of Human-Computer Studies 59 (2003) 81-118

[12] Nicholson, A. and Brady, J.: Dynamic belief networks for discrete monitoring. IEEE Transactions on Systems, Man, and Cybernetics 24 (1994) 1593-1610

[13] Rao, A.S. and Georgeff, M.P.: BDI-agents: from theory to practice. In: Proceedings of the 1st International Conference on Multiagent Systems (ICMAS '95), San Francisco, USA (1995)

[14] Duffy, B.R., Dragone, M., O'Hare, G.M.P: Social Robot Architecture: A Framework for Explicit Social Interaction. Android Science: Towards Social Mechanisms, CogSci 2005 Workshop, Stresa, Italy (2005)

[15] Konolige, K., Myers, K., Saffiotti A. and Ruspini, E.: The Saphira architecture: a design for autonomy. Journal of Experimental and Theoretical Artificial Intelligence 9 (1997) 215-235

[16] Ulrich I. and Borenstein, J.: VFH+: Reliable Obstacle Avoidance for Fast Mobile Robots. IEEE International Conference on Robotics and Automation, Leuven, Belgium (1998)

[17] Fox, D., Burgard, W. and Thrun, S.: The dynamic window approach to collision avoidance. IEEE Robotics and Automation Magazine 4/1 (1997)

[18] Ross, R., Collier, R. and O'Hare, G.M.P.: AF-APL: Bridging princples & practices in agent oriented languages. In: Proceedings of the 2nd International Workshop on Programming Multiagent Systems: Languages and tools (PROMAS 2004). Held at AAMAS '04, New York, USA (2004.)

[19] Collier, R.W., Agent Factory: A Framework for the Engineering of Agent-Oriented Applications, Ph.D. Thesis, University College Dublin, Ireland, 2001

[20] Rooney, C F.B., Collier, R.W., O'Hare, G.P., VIPER: VIsual Protocol EditoR, in 6th International Conference on Coordination Languages and Models (COORDINATION 2004), Pisa, February 24-27, 2004.

[21] Coradeschi S. and Saffiotti, A: Perceptual Anchoring of Symbols for Action. In: Proceedings of the 17th IJCAI Conference, Seattle, USA (2001)

[22] Kuipers , B.: Qualitative Simulation. Artificial Intelligence 29 (1986) 289-338

[23] Sheshagiri M. and des Jardins, M.: Data Persistence: A Design Principle for Hybrid Robot Control Architectures. Proceedings of the International Conference On Knowledge Based Computer Systems, Mumbai, India (2002)

[24] Duffy, B.R., Garcia, C., Rooney, C.F. and O'Hare, G.M.: Sensor Fusion for Social Robotics In: Proceedings of the 31st International Symposium on Robotics (ISR '00), Montreal, Canada (2000)

[25] Searle, J.: Speech Acts: An essay in the philosophy of language. London, Cambridge University Press (1969)

[26] Duffy, B.R.: Social Embodiment in Autonomous Mobile Robotics. International Journal of Advanced Robotic Systems 1/3 (2004) 155-170

[27] Kato, H., Billinghurst, M. and Poupyrev I.: ARToolKit. Technical report, Hiroshima City University (2000)