2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology

# An Affective Anticipatory Agent Architecture

David Sanderson
*ISN Group, EEE Dept., Imperial College*
*Exhibition Road, London, England, SW7 2BT*
*dws04@imperial.ac.uk*

Jeremy Pitt
*ISN Group, EEE Dept., Imperial College*
*Exhibition Road, London, England, SW7 2BT*
*j.pitt@imperial.ac.uk*

***Abstract*—We present an agent architecture for an enhanced theory of intent prediction with affective evaluation of expectations. The architecture combines models from psychology and robotics to create an online situation appraisal mechanism for preferential evaluation of predicted future states, which determines action selection in the current state. The architecture is implemented in a vehicular agent situated in a motorway driving scenario, requiring more than goal-directed planning: the agents must model the behaviour of others, and predict and evaluate future states. Simulations are described, showing that global properties emerge in the system that are improved and more stable with the new agent architecture.**

***Keywords*-Pro-activity; Expectation; Anticipation; Affective**

## I. Introduction

Wooldridge and Jennings wrote in [1] that an agent could be informally defined by its possession of the following properties: autonomy, social ability, reactivity, proactiveness. The first three concepts are relatively straightforward, but the concept of 'pro-activity' is not simply reactive, "goal-directed behaviour"; it is instead a subtle, nuanced, concept covering a spectrum of behaviour requiring the agent to model other agents, anticipate future states and actions, and perform cognitive evaluations of their quality [2].

To develop an operational property of pro-activity, we propose to combine a theory of intent prediction based on aspects of the HAMMER (Hierarchical Attentive Multiple Models for Execution and Recognition) architecture [3] with an affective evaluation of predictions [2]. This will allow us to fully develop a method of intent prediction based on affective appraisal of possible future states.

The scenario in which our experiments take place will serve as a running example in this paper. We have considered a simplified motorway carriageway consisting of three lanes of traffic all moving in the same direction. The driving scenario will require the vehicles to attempt to fulfil their goals - which lane to drive in, what speed they should travel at, and how much space to leave between themselves and other vehicles. This implies behaviours that the vehicles will need to exhibit; as well as continuing at their current speed, they must be able to select from the actions "Increase Speed", "Decrease Speed", "Change lane left", and "Change lane right". In the course of moving through this simulation an agent may find itself faced with certain situations requiring it to reason about the current state, or possible future states, and these actions will allow them to overtake and avoid other vehicles whilst maintaining their own progress.

In order to achieve pro-activity, our model will need to contain both a mechanism for predicting the future states and a method of evaluating these predicted states with regards to the preference of the agent. In the next section we present HAMMER as the basis of our predictor and a cognitive functioning of expectations as the affective dimension of the evaluation of states. Section III describes the architecture of our model in detail. The results of the simulations performed are set out in Section IV and Section V summarises the work and presents some possible future work.

## II. Architectural elements

### A. HAMMER

The HAMMER architecture is a framework for online action selection and intention recognition developed for the robotics and multi-agent systems domains that uses the *generative*, *simulationist* approach to understanding actions by mentally rehearsing them from a motor-based perspective [3]. It uses Inverse-Forward model pairs with an additional "Prediction Verification" block, as seen in Fig 1, in parallel and then feedback/competition to select the correct action. We use HAMMER as both a "controller" to plan and control the vehicular agent, and also as a "predictor" to make predictions about the actions of the neighbouring vehicles, that are then fed into the planning phase of HAMMER.

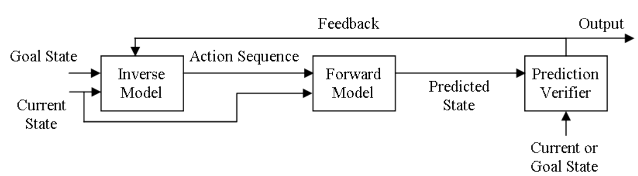


Figure 1. A Model Pair in HAMMER

### B. *Affective evaluation*

The cognitive anatomy of expectations and their function in purposive action in [2] presents an analytical decomposition of the structure of expectations, as well as their purpose,

978-0-7695-4513-4/11 $26.00 © 2011 IEEE
DOI 10.1109/WI-IAT.2011.178

meaning, and side-effects arising from their fulfilment or invalidation. We use this to inspire the affective evaluation of predicted states.

In the work, an "Expectation" consists of two parts; a *Belief* with strength and a *Goal* with value. These are the subjective certainty and importance of the expectation respectively. We base our method of affective evaluation on the concept of an expectation having a value to the agent and a certainty weight. As an agent moves through the world it will have to make choices about its actions and be able to rank its goals. Given a goal $G_i = \langle V_i, W_i \rangle$ with value $V_i$ and weight $W_i$, the expectation associated with its fulfilment $E_G(G_i)$ is given by (1) where $A$ is the actual value of the variable the goal relates to, $G_i \downarrow_1$ and $G_i \downarrow_2$ are the value and weight of $G_i$ respectively, and zero is total satisfaction.

$$E_G(G_i) = \frac{G_i \downarrow_1 - A}{G_i \downarrow_1} G_i \downarrow_2 \tag{1}$$

Once the goals have been ranked, an agent may have to choose between a number of actions that will each allow it to move towards a different macro-level goal. In order to make this decision a 'MicroGoal' $M_i$ is formed as in (2) where $\widehat{x}$ is the normalised value of $x$.

$$M_i(G_i) = \left\langle G_i \downarrow_1, \widehat{E_G(G_i)} \right\rangle \tag{2}$$

Finally, an agent will need to rank the various states it has predicted. The expectation $E_S(S_i)$ associated with any state $S_i$ is given by (3) where $C(S_i)$ is the relative confidence the agent feels about the likelihood of $S_i$ coming to pass.

$$E_S(S_i) = C(S_i) \sum_i \{M_i(G_i) \downarrow_2\} \tag{3}$$

## III. Intelligent agent design

The design of our agent architecture is shown in Figure 2 and we will use the running example of a motorway driving scenario to ground its description.

An agent will always require a world to sense and act upon. This is represented as $W(Ag_i, t)$ for an agent $Ag_i$ at time $t$ as in (4) with the agent's own state as $State(Ag_i, t)$, the agent's neighbours as $\eta(Ag_i, t)$, and everything else in the environment as $Env_t$. The agent's state will need to combine its "characteristics" or "personality" (which may be mutable) and its state inside the world (which will be mutable). Our simplified motorway environment has discrete lanes represented as a 3x$length$ array, with each vehicle's position represented by a $(lane, position)$ coordinate.

$$W(Ag_i, t) = \langle State(Ag_i, t), \eta(Ag_i, t), Env_t \rangle \tag{4}$$

A purposeful agent will always require goals; in our model the agents have their overarching 'MacroGoals', which consist of a number of `AgentGoal`s. There are $(value, weight)$ pairs with an `Integer` value representing what the actual goal is, and a `Double` weight representing how important this goal is to the agent. We consider three goals: `speed` (the agent's preferred speed), `spacing` (the minimum preferred spacing between the agent and any other agent), and `lane` (the agent's preferred lane).

To affect its world and pursue its goals, an agent will require a set of actions. The scenario informs the actions we require and they are shown in Table I. The agent is designed with high-level actions as this is only a demonstration of the cognitive functions of the agent, not its ability to drive. The `action0` can be thought of as a "no action" choice in that the agent will not manoeuvre, but it also contains collision-avoidance measures. For example, if there is an obstacle ahead the agent will overtake the obstacle unless this would also cause a collision, in which case it will slow down.

Table I
Agent Actions

| Action | Description |
|---|---|
| 0 | Continue at current speed in current lane. |
| 1/2 | Change lane up/down. |
| 3/4 | Decelerate : reduce speed by one/two. |
| 5/6/7 | Accelerate : increase speed by one/two/three. |

In the "Assess Situation" module, the agent's own state is used to evaluate its adherence to each of its MacroGoals using Equation (1); for example if the agent is travelling at a speed of 4 and its `speed` MacroGoal is 5, then it would result in a score of 0.2. These scores are multiplied by the weight of the corresponding MacroGoal and normalised, resulting in a ranked list of 'MicroGoals'. A MicroGoal is a kind of `AgentGoal` as described above. If a MacroGoal is being completely achieved then the corresponding MicroGoal will have a weight of zero.

This list is used as the input to a simplified inverse model in the "Goal Selection" module. This module represents the internal knowledge that an agent would have about the world it inhabits. Its purpose is to pass a list of actions to the HAMMER module that are both physically possible and ranked in order of probable usefulness in achieving the agent's goals. A cause-effect relationship between action and result can be used to generate an action for each MicroGoal; for example speeding up or slowing down is likely to have more of an effect on the spacing between vehicles than changing lane will. Note that it is possible that an agent might be fulfilling all of its MacroGoals fully and therefore have no MicroGoals with non-zero weight. In this case, the module will simply return a single `action0` action so that the agent will continue on as it is.

Separately to the list of MicroGoals being processed, the agent's neighbours are retrieved; a HAMMER "prediction" process is run for *each* of the agent's neighbours to predict their possible future states. This process involves generating a ranked list of MicroGoals and actions as above, and for

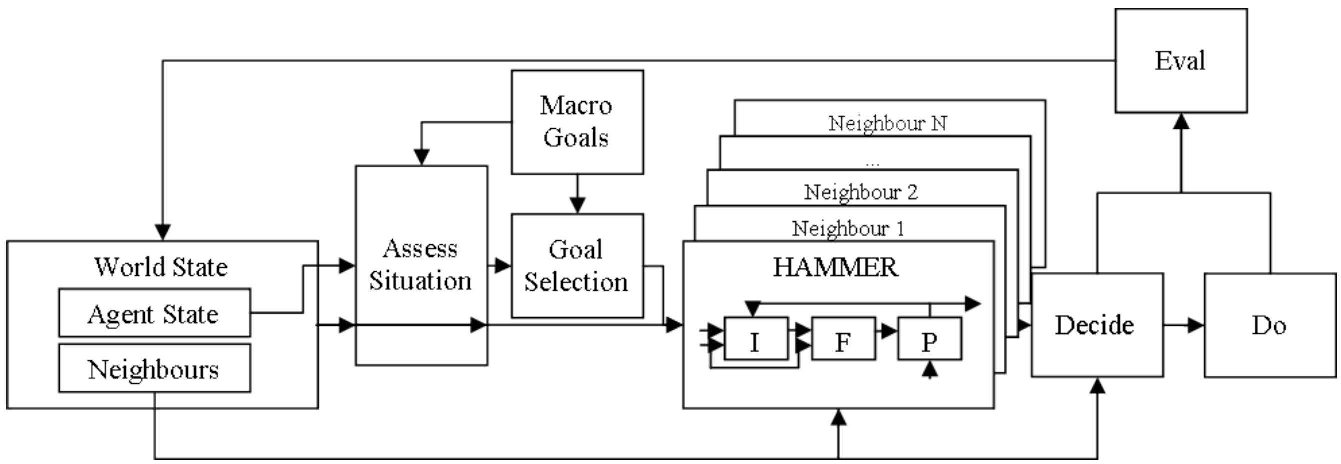


Figure 2. Agent Architecture diagram

each action simulating the outcome. Each of these outcomes is ranked (in the same manner as the "Assess Situation" module, also see Equation (3)) and the best outcome is chosen for each neighbour. Each of these 'best outcomes' is added to a set of predicted future world states.

Using the predicted world, each of the possible actions (that the agent generated in the first stage) are simulated using a HAMMER "planning" process and ranked. This will generate a number of predicted possible future world states from which the agent can choose the preferred one.

## IV. Experimental results

The experimental set-up is defined on a finite series of agent models $M_t = \langle Ag, Arr, Act, Env \rangle_t$ where $Ag$ is the set of agents whose arrival rate $Arr$ is uniform but locally random, and $Act$ is the set of physical actions they can perform (Table I) in physical motorway environment $Env$.

Each agent is generated with random macro-level goals and parameterised by $Ag_i = \langle G, H, E_S\left(S_i\right)\rangle_i$ where $G$ is a set of $(value, weight)$ goal pairs for preferred speed, lane, and distance (as specified in Section II-B), $H$ is the action selection function such that for each agent $i$ at each time slice $t$, $H_i(Env_t) \in Act$.

At each time slice, the metrics collected include each agent's position, speed, and $E_S\left(S_i\right)$, and the system's *congestion* (distribution and number of active agents) and *throughput* (number of agents finished).

### A. Baseline vs pro-active agents

An experimental baseline was established by running a number of simulations for agents with no predictive or affective component; the $H_i\left(Env_t\right)$ selects the first action that would result in a valid state (the agent does not crash) while keeping at a constant speed and lane. Simulations were then run for the same generative data ($Act$, $Env$, $Arr$, $G_i$ are unchanged) for the same number of runs, but this time with $H_i(Env_t)$ as the HAMMER-Castelfranchi action selection mechanism described in Section III. Each run for the pro-active agents takes around four hours on a $3GHz$ AMD with $8GB$ RAM; the runs shown are representative.

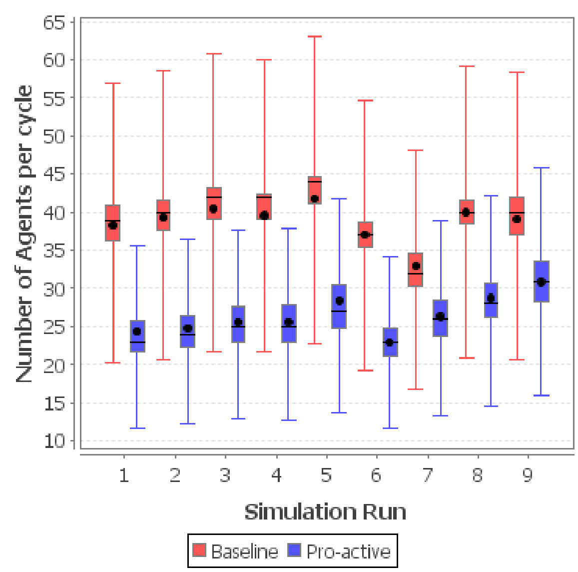


Figure 3. System congestion

### B. Analysis

The *throughput* of the system is the number of agents that have finished after each cycle; as expected, the *throughput* for the 'baseline' and 'pro-active' agents tends to the same gradient because the arrival rate is the same in both cases and "what goes in must come out" given enough time.

Figure 3 shows the *congestion* of the system; each plot represents how the number of active agents in each cycle varies over a given run. The line and dot are the median and mean respectively, with the box showing the moving standard deviation, and the whiskers showing the moving average. An average improvement of around $40\%$ is observed.

Although the *throughput* of the system will always tend to the same value given enough time, the *congestion* is an emergent property of the system and is affected not only by the speed of each agent but also by the formation of clusters in the population as one slow-moving vehicle causes others to slow down behind it, or to overtake. The pro-activity afforded by our architecture allows the 'pro-active' agents to avoid situations where they may become congested, and this in turn "smooths out" the clusters observed in the population of 'baseline' agents. Due to the varying speed goals of the agents, there may still be some slow-moving vehicles grouped together that cause a cluster to form even with the 'pro-active' agents. We postulate that this is attributable to a lack of communication and teamwork and for this reason we propose to extend the framework to investigate the effect of pro-active agency in a model of teamwork.

When comparing the dissatisfaction of the agents given by $E_S(S_i)$, we find that the addition of pro-active behaviour improves the comfort of the agents by allowing them to more effectively avoid undesirable situations and attain desirable ones.

## V. Summary and conclusions

The research presented in this paper has resulted in a novel method for reasoning about other agents' intentions with respect to Beliefs and Desires without using the BDI Model of agents, using instead the HAMMER architecture from the field of robotics. The affective evaluation of current and predicted states that is required for pro-activity has been accomplished by integrating a formalisation of expectation into our architecture.

By showing that our agent improves and stabilises system congestion and global agent happiness, we conclude that there is a social utility in exhibiting affective, anticipatory reasoning and that integrating an affective evaluation of predictions into an intelligent agent architecture provides significant global advantages. A formal characterisation of a theory of pro-activity is essential for further experimentation in teamwork. It is our intention to extend our work to cover multiple agents cooperating and/or competing as groups. Any group of agents would have a set of social normative rules to govern their behaviour; this can be investigated with respect to the transportation setting, as well as the interesting area of anticipatory reasoning about other agents' intentions within a set of norms. In general a driver wishes to get from their start point $A$ to their destination $B$ in a given time $t$, with $zero$ accidents, and obeying a set of additional rules $R$ that may be personal (such as fuel efficiency or a certain route) or external (such as laws or relevant social norms). This formulation can replace the set of simple instantaneous goals used so far.

Despite being initially developed as an explanatory *theory* for the intentional stance in humans [4], the BDI concept has come to be seen as a *model* specification for software agents that are to reason about intentions. This results in attempts being made to directly add things to the model (such as emotions and obligations) that simply were not present at all in the original theory. By separating the affective evaluation from the computational model, our model allows emotions to be expressed in any suitable terms, rather than in terms of (in our case) the HAMMER model. This means that we can integrate emotions with respect to the principles of both theories; the explanatory adequacy of the affective evaluation has not compromised, or been compromised by, the operationalisation of the HAMMER architecture. Our model retains conceptual clarity, expressive capacity, computational tractability, and predictive leverage without adversely affecting any of them.

We propose a multi-dimensional approach to intelligent transportation. An intelligent transportation system can be seen as being composed of three tiers; the micro-level of each vehicular agent, the meso-level of groups of vehicles, and the macro-level infrastructure. The work that has been accomplished so far provides the first step towards this by developing an intelligent anticipatory vehicular architecture to serve as the micro-level of the system. Any group of agents would have a set of social normative rules to govern their behaviour; this can be investigated with respect to the transportation setting, as well as the interesting area of anticipatory reasoning about other agents' intentions within a set of norms. Roughly, this corresponds to the meso-level of the three-tier system described.

## Acknowledgment

This research has been conducted with the support of an EPSRC Studentship and an IBM Faculty Award.

## References

[1] M. Wooldridge and N. Jennings, "Intelligent agents: Theory and practice," *Knowledge engineering review*, vol. 10, no. 2, pp. 115–152, 1995.

[2] C. Castelfranchi and E. Lorini, "Cognitive anatomy and functions of expectations," *Proceedings of IJCAI03 Workshop on Cognitive Modeling of Agents and Multi-Agent Interactions*, pp. 9–11, 2003.

[3] Y. Demiris, "Prediction of intent in robotics and multi-agent systems." *Cognitive processing*, vol. 8, no. 3, pp. 151–8, 2007.

[4] M. E. Bratman, *Intention, plans, and practical reason*. Harvard University Press Cambridge, MA, 1987, vol. 250.