

TUNING BEHAVIOURS USING ATTRIBUTES EMBEDDED IN AN AGENTS' ARCHITECTURE

José Cascalho ^a

Helder Coelho ^b

^a *Universidade dos Açores 9701-851 Angra do Heroísmo Portugal,
jmc@notes.uac.pt*

^b *Departamento de Informática Faculdade de Ciências da Universidade de Lisboa
Bloco C6, Piso 3, Campo Grande 1749-016 Lisboa, Portugal, hcoelho@di.fc.ul.pt*

Abstract

In this paper we discuss how to *tune* agents' behaviours by explicitly modify a set of elements previously defined and included in the agents' architecture. By *tuning*, we mean influencing agents' world view, changing their preferences and even modify their beliefs about which goals are possible. This elements which we call attributes, such as urgency, insistence and intensity, are able to modify agents' priorities with regard to the resource consumption, to modify the evaluation of the implicit costs of action execution and even to change agents' view about their capabilities to execute an action.

In a preliminary experimental evaluation made in a multi-agent system environment, a modified predator-prey workbench, we show how the attributes are important elements while trying to improve predators' global efficiency.

In the final discussion, we argue about the benefits of having this set of attributes which allow agents to be selected and modified by stakeholders to support environmental odds, like a team manager would do.

1 Introduction

In this work we discuss the use of attributes to characterise different agents behaviours. These attributes, intensity, insistence, importance and urgency are embedded in the agents' architecture and we use them to change the agents' decision process. For example, the attribute insistence measures how much effort an agent is capable to do to achieve a goal. A high value of insistence means that an agent persists in achieving a specific goal during a predefined time.

These attributes come from previous work [3][5] and have been already used on trying to modulate the decision process in cognitive agents [15][16].

In this paper we explore these attributes of the agents' architecture in a predator-prey environment [11]. We create predators with different values of those attributes and test the predators' global performance, by measuring the average time the predators stay alive in the world.

We argue that these different behaviours that can be created using different attributes (characterised by environment properties where agents belong), can be used to define *agents' types* that could help a stakeholder to select the right set of agents among a predefined set, as a way to improve system's performance.

Our thesis is that agents, like humans, need to be different (or behave differently) to tackle different situations like an emergency where it is important an energetic response versus a trading where there is a need to evaluate carefully the different proposals from different traders.

2 The Agent's architecture

Our agents form a goal governed-system [6], that is, the agents have goals explicitly represented which they want to satisfy. We are interested in studying the agents' inner mechanism for selecting behaviours.

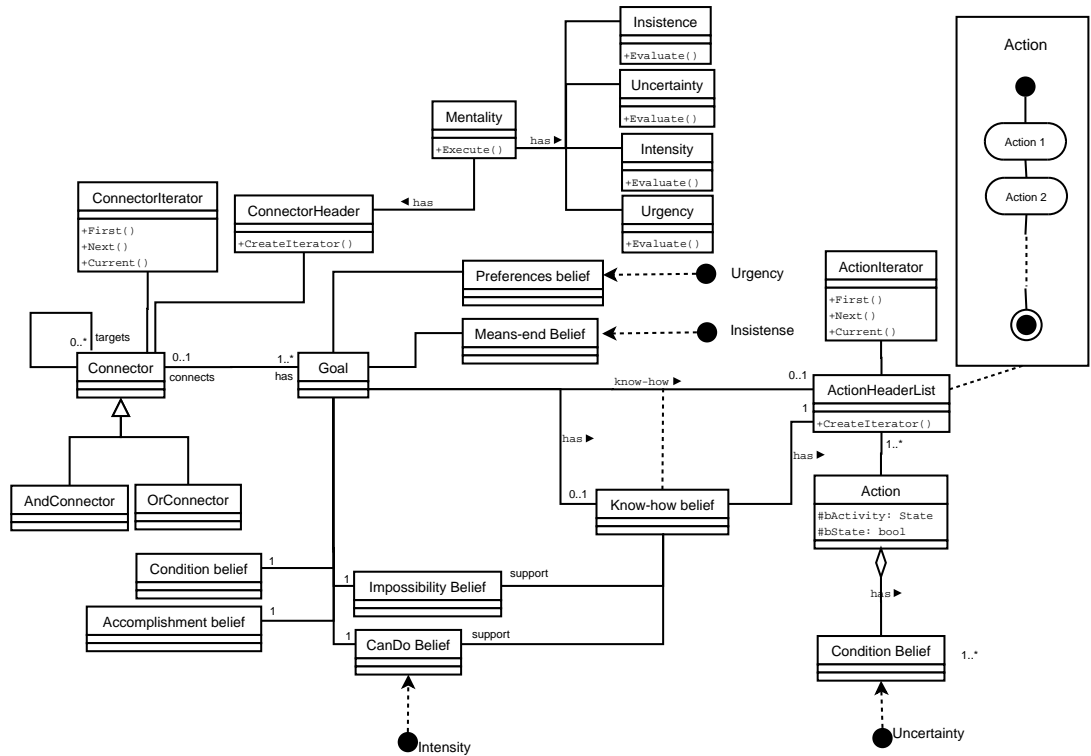


Figure 1: The agent's architecture.

The agents' mind architecture has a set of goals which are linked by an OR-connector or an AND-connector (AND/OR decomposition [17]). These connectors allow the agents' designer to create a tree (AND/OR tree) which makes possible to define 'and-goals' (the children of an AND-connector node) or 'or-goals' (the children of an OR-connector). Both constitute alternative *paths* to satisfy a goal. The former means that a goal can be satisfied if all the and-goals are executed successfully (represents problem decomposition into sub-problems all of which must be solved), while the latter means that to satisfy the goal, it suffices only one of the or-goals be satisfied.

In figure 1 a schematic representation of the architecture shows the different classes and their main connections. Next, we explain succinctly some of the beliefs supporting the agent's decision process:

- The **accomplishment belief** defines the conditions in which a goal is satisfied.
- The **know-how belief** links a goal to an *Action*, i.e. a *plan* to satisfy a goal. In our architecture this corresponds to a sequence of atomic actions.
- The beliefs **condition belief** defines the (pre-)conditions that support the execution of the goals and the actions of a *plan*.
- The **cando belief** evaluates the internal agents' capabilities. The difference between a *condition belief* and a *cando belief* is that the former is part of the goal (or action) external condition to the goal's (or action's) execution while the latter is an evaluation of the agents' ability (an agent can have the conditions but not the ability to reach a goal)¹. The attribute **intensity** is evaluated with respect to the agents' ability to execute or not execute a goal.

¹Both beliefs evaluate if the goal can be satisfied if the action associated to the *know-how belief* is executed.

- The **preferences belief** defines the order in which a sub-goal is chosen. The **urgency** influences this order.
- The **means-end belief** test the conditions for a goal execution (see below for a detailed explanation of the role of this belief).

Let's now describe the main cycle (*reasoning-cycle*) of the agent's mind. The agent keeps a pointer to the goal that is executing. The agent starts to select the goal in the top of the tree, the start node. In the **means-end belief**, a test is made to evaluate whether a goal is satisfied or not by calling the **accomplishment belief**. If it is satisfied, he looks for another goal in the AND/OR tree. Otherwise, he checks if the **condition belief** is true and verifies if there is a plan to execute the goal (**know-how belief**). Finally the agent evaluates the **Impossibility belief** and the **Cando belief**. If all conditions return true, he selects an action to execute, following the plan attached to the goal.

An agent search for a new goal in the AND/OR tree when there is at least one condition, from the conditions enumerated above, returning false.

The leafs or terminal nodes (having no child nodes) in the goals-tree might have variables which must be instantiated. If this is the case, inside the **means-end belief**, an instantiation policy is used to order the list of possible instantiations. This instantiation policy is supported by **preference belief** which orders the options based on a predefined criteria ².

After each *reasoning-cycle* in which the mental state is updated, an agent will execute an action (possibly a NULL action).

3 Role of attributes

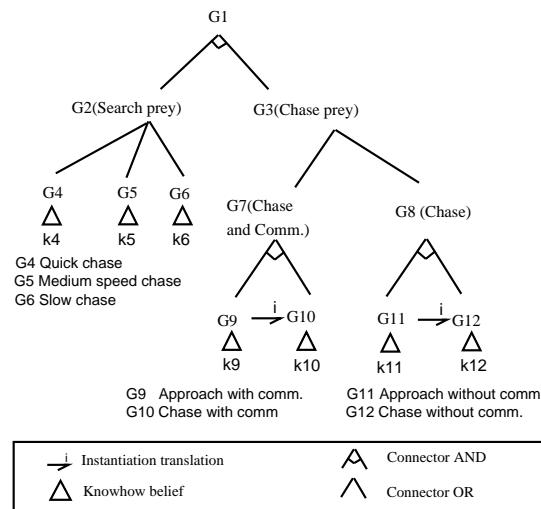


Figure 2: The AND/OR tree of goals.

In figure 2 we present the goal-tree we used in the experiments. An agent walks through the goal-tree looking for a goal not satisfied and *executable*. When he has the conditions to achieve the goal, he uses the plan attached to that goal and executes the actions in the plan. An evaluation of success is kept. When a goal succeeds (the accomplishment belief returns true) he stops the searching activity and, in the next cycle, restarts again the search from at the start node of the tree. If the search for a the conditions to select a goal fails, the agent returns a null action.

Intensity measures the willingness of the agents to spend their energy when satisfying a goal. Energy is a resource, so we can say that intensity measures how much of a resource an agent is willing to spend to satisfy a goal. In the meta-model we link the intensity attribute to the belief 'CanDo'. This means that to each goal's plan and possibly to each action inside that plan, an

²The instantiation policy is related to the attribute importance which is not discussed in this paper.

evaluation is made about the possibility of executing that plan or action with respect to agent’s ability to spend that resources. Note that ‘intensity’ doesn’t judge about the worthwhile of an action. Agents get a constant value of intensity and this will characterise their behaviour.

Insistence attribute is added to the agent’s means-end analysis in the architecture (see figure 1). For each goal, an agent defines a plan as a sequence of steps to accomplish a goal. It measures the agents persistence toward a goal, i.e. how much time an agent will try to satisfy a goal. If after a number of attempts a goal is not yet accomplished, the agent reconsiders the strategy to accomplish that goal, marking as a failure the previous strategy and searching for another strategy. Agents receive a constant value of insistence and this will characterise their behaviour.

The urgency acts as a regulator of agents behaviour. A set of context variables are used to calculate the urgency value and are usual related to distressed situation. With the increase of the urgency the agents will change (amplify or contract) the effects of the intensity or insistence (i.e. with urgency equal to 0.9, an agent with a value of intensity equal to 0.5 could behave as an agent with a value of intensity equal to 0.8). Instead of being defined as a constant value, urgency is dynamically dependent of environment parameters. These parameters can change also dynamically which means that an agent is able to adapt in real time to changes in the environment.

4 Experimental setup

We adapted a pursuit simulator [11] by adding the *energy* resource to the predators. This new parameter changes the way the simulator ends a game, i.e., a game ends after the death of all predators³ before they could catch all preys launched in the game’s beginning. When a new game starts, the energy is restored to initial values. In the original simulator, an unspecified number of predators tries to catch one or more prey agents. A game in the simulator is defined by episodes and cycles. In each cycle the simulator receives information through sockets about the moves of predator and prey agents, and messages exchanged among predators. An episode ends when all prey is caught and a new episode starts with all predator and prey agents randomly repositioned in the field. Data about how long agents take to catch all prey agents are kept as a statistical measurement of predator efficiency.

The pursuit domain was introduced in [2]. It has been widely used testbed for multi-agent systems. Several variations of the original descriptions have been studied over the years (see [10] for more details). The domain we used in our experiments consists of a discrete, grid world of 20X20, in which the predators catch a prey when predator and prey share the same cell (one of the predefined capture criteria in the simulator [11]). Predators and prey can move to north, south, east and west or stay in their position. They consume more energy when they move than when they maintain their position. They can send messages to each other⁴. Finally, their total energy is incremented by a certain predefined amount after they catch a prey.

In the experimental environment the density of prey agents is equal to 0.03 (12 preys per 400 places) and the number of predators corresponds to 1/4 (4 predators for 12 prey) and 2/3 (8 predators for 12 preys) of the total of preys. To measure the global performance of all the agents, for each experiment we record the number of episodes they survive. We run the simulator about 80 times per experiment, i.e. for different set of agents, we run about 80 times the workbench.

5 Experiments description

In the next sections we explain the different experiments we made to test the different attributes, intensity, insistence and urgency, under the pursuit domain (see figure 3) . We divided our experiments up into two different parts. In the first part we evaluated the attributes intensity and urgency associated to the prey search. Those experiments are related to the selection of the goals G4, G5 or G6 of the goals-tree presented in figure 2. Four predators chased twelve random preys (the preys select randomly one of the possible movements, north,south, east, west or don’t move). Selecting behaviours controlled by the attributes urgency and intensity contributed to the improvement in

³They die after their energy decreases below a survival threshold.

⁴In this setup, an agent doesn’t consume energy by sending messages

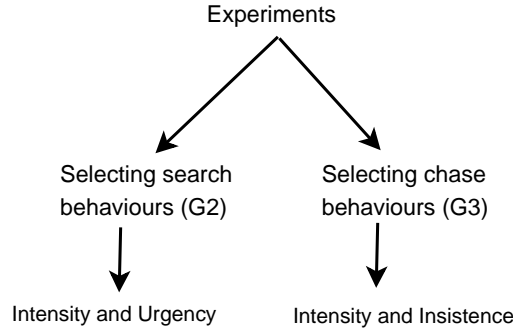


Figure 3: The experiments.

50% of the global system performance⁵. In the second part, we tested the attributes insistence and intensity in a new setup where preys, instead of moving randomly, run away from the predators but with different *speeds*⁶. In this part of experiment we used the right part of the goals-tree, the goals G9, G10 and G11, G12 pairs.

5.1 Searching preys with attributes intensity and urgency

An **intensity** measures the ‘potential’ an agent applies to satisfy a goal [12]. We link this attribute to the resource consumption. Then an *high intensity agent* (IA) is a predator that **selects** a strategy that corresponds to a high energy consuming behaviour. The agent selects the goals differently based on the energy he has and his value of the intensity attribute. This selection is made by the ‘Preference belief’ following the premises:

- It is given to an agent a certain level of intensity **i**.
- To the goals G_4, G_5 and G_6 correspond plans with different levels of energy consumption per time step (**ets**), where G_4 has the highest consumption and G_6 has the smallest.
- The agent divides ets_{G_4} by his total energy, multiplies by **i** and if the consumption is in the interval above selects G_4 :

$$0\% \leq \mathbf{i} * ets_{G_4} < 5\% \rightarrow G_4$$

- It do the same for the other two goals, with the percentages and following the order of the expressions above:

$$5\% \leq \mathbf{i} * ets_{G_5} < 10\% \rightarrow G_5$$

$$10\% \leq \mathbf{i} * ets_{G_6} < 50\% \rightarrow G_6$$

- Finally he doesn’t move if energy consumption is above 50% for the less consumption goal:

$$\mathbf{i} * ets_{G_6} \geq 50\% \rightarrow None$$

Using two parameters from the context, the optimum value for the number of preys to be caught and the average capture time of all preys in the episodes, the agent calculate the value of urgency to satisfy the goal ‘searching a prey’.

The urgency value is calculated using the following expression:

$$urgency = weight * (bias + 1 - Nr.PreyCaught/PreyTarget) +$$

⁵see [4] for a fully description of this experiment

⁶We define three type of preys, quick, normal and slow preys, to which we ascribed different number of moves per cycle rate

$$(1 - weight) * (bias + Nr.Cycles/CyclesAverage - 1),$$

where *bias* gives a ‘value of reference’ for the urgency when the number of prey caught is equal to the value *PreyTarget* and the the number of cycles in the episode is equal to the value *CyclesAverage*.

Our goal was to allow the agent to select the goal that reveal the greatest efficacy. While applying the attribute urgency to the *preference belief* the agent do the following:

- If the urgency is low, the agent tends to save resources and so it will use the strategies with which the energy/steps rate is smaller.
- If the urgency is high, the agent must use all the resources he has to quickly solve the problem he has at hand, so he admits to select the best strategy even if he has to spend the rest of the resources he has got.

The scenario with the attributes urgency and intensity increased the agents’ performance by 50% when compared with the scenario with just the attribute intensity [4]. Urgency rationalizes an energy consumption by letting agents select behaviours with higher consuming rates only in urgency situations.

5.2 Chasing preys with intensity and insistence attributes

In these experiments we created new preys which runaway from predators. We add three type of preys each one with different speeds (average of the number of moves per time step ⁷).

The agents not only choose the preys to catch but also select the direction they use when approaching preys. The preferences belief will determine which prey is selected, in this case the nearest prey⁸, i.e. an agent instantiates the goals ‘approach prey’ and ‘chase prey’ giving values to these two parameters. For example, an agent can select the nearest prey with number ‘1’ and with ‘north chase direction’.

Furthermore, agents have two sources information, visual and agent’s messages. A predator, when chasing a prey (goal G10), sends a broadcast message with information about his relative position to the prey he is chasing. When the other agents receive this message they check if they see the predator who sends the message and if this is the case they calculate the position of the prey he is chasing⁹. This increments the number of preys each agent can see.

The insistence measures the persistence an agent has toward a goal. After selecting a prey to chase, a predator chase that prey a maximum number of cycles (parameter *chase_prey_max_cycles*). If he doesn’t catch the prey within this time interval, he gives up from catching that prey and tries to catch another one. Besides, when an agent gives up, he inserts the prey number in a list of failures, *the preys’ dark list*). Another parameter (*dark_list_time_interval_prey*) defines how much time the information about a failure is kept in that list. While this information is in the list the agent won’t chase that prey again.

The agent uses the following expression to calculate the maximum time he chases the same prey,

$$chase_prey_max_cycles_{prey} = distance_to_prey * e^{insistence*k}$$

and the expression to keep information about the failed attempt to catch a specified prey,

$$dark_list_time_interval_{prey} = max_interval * insistence^k * e^{1-insistence^k}$$

The parameter *max_interval* is constant with value 40. The parameter *k* is equal to 3 or 4 (k3 or k4). These values were selected to allow the values of *chase_prey_max_cycles* to be between 1 and 208 cycles (*k* = 3) or between 11 and 512 cycles (*k* = 4) and the values of *dark_list_time_interval* to be between 0 and 40 cycles.

Finally, in this experiment, the intensity is used to decide if agents chasing a prey broadcast a message with the information about its relative location (selecting between the branches G7 or

⁷All preys moves slower than predators, but among them there are 4 quick speed preys, 4 slow speed preys and 4 with a speed between the others.

⁸This decision is part of *agents’ policy* which can be changed, but this is not discussed in this paper

⁹There isn’t an absolute position reference in the world, so predators only determine the prey position relative to their own position.

(Nr.Pred)xInsistence	Mean ($k=3$)	Mean ($k=4$)
(8)x0.3	14.3	-
(4)x0.3 (4)x0.6	19.2	-
(8)x0.6	18.1	23.2
(4)x0.6 (4)x1.0	16.9	-
(8)x1.0	21.2	8.0

Table 1: Average of the number episodes per game, for a total of 70 games with $k=k3$ and $k=k4$. All the five experiments had 8 predators and 12 preys.

(Nr.Pred)xIntensity	Mean	Standard deviation
(8)x0.5	8.27	6.55
(8)x0.8	13.31	14.78
(8)x1.0	21.2	19.3

Table 2: Average of the number episodes per game for 70 games. All experiments had 8 predators, with insistence 1.0 and intensity equal to 0.5 and 0.8.

G8 of the goal-tree). With the intensity equal to 1.0, it is certain that they broadcast the relative position of all the preys they chase. The probability of doing it decreases with the decreasing value of intensity.

5.3 The effect of insistence

Several experiments were made with different values of insistence. The table 1 summarise the results. In the table we verify that the best result is obtained for insistence equal to 1.0 and $k = 3$, for all 8 predators (see last row in the table). We were expected that the effect of insistence could improve the performance because this attribute gives to agents the capability to chase a prey for a large amount of time. On the other hand, we suspected that if an agent *had too much* insistence he could prejudice his global performance because he would eventually lost almost all of his energy trying to capture a quick prey. So we changed the parameter k from k3 to k4 (last column in the table) extending the time the agent would chase a prey without quitting. And we found that now with insistence equal to 1.0, the mean become 8.0. Comparing the means, we also observe that the insistence equal to 0.6 is now the best result and surprisingly surpass the previous value for k3 and insistence equal to 1.0, but supporting our suspicion.

When changing k3 to k4, the predators never quit chasing preys after they fix them. They will only stop chasing one prey if the prey disappears (for some reason they lost their trace) or if the prey is caught. The figures 4 ,5 and 6 show the number of times a predator quit chasing a prey because he surpasses the threshold defined for the insistence. We notice that for insistence 1.0 with k4, predators seldom quit chasing preys. A quite different scenario is observed for insistence 1.0, k=k3 and for insistence 0.6, k4 (figures4 and 5respectively).

5.4 The effect of intensity

In the table of figure 5.4 the results for different values of intensity are presented. We notice that the results get better as the intensity gets higher. We conclude about the importance the agents' messages about the position of preys being chased have in this specific experimental setup.

6 Conclusion

In predator-prey scenario, the use of attributes becomes extremely useful in searching for good solutions. Intensity, insistence and urgency were tested and their values influenced significantly the global performance of the system. This open alleys for a deeply research of the possible types

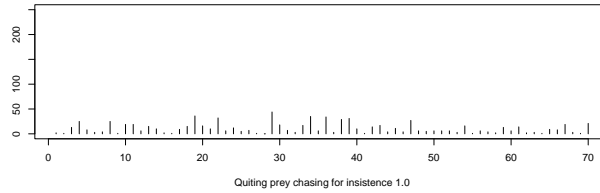


Figure 4: Nr. of times predators quit chasing preys without catching them (insistence=1.0, k3).

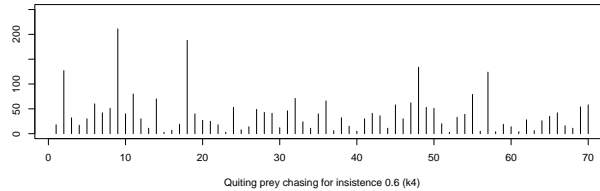


Figure 5: Nr. of times predators quit chasing preys without catching them (insistence=0.6, k4).

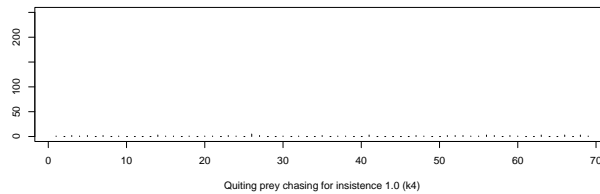


Figure 6: Nr. of times predators quit chasing preys without catching them (insistence=1.0, k4).

of agents that could be defined with help of these attributes, and also to extend this results for different applications where a definition of the same attributes could be implemented.

Indeed to a stakeholder several properties must exist and should be maintain in a system, especially when in presence of some distressed situations. Along with the usually desirable performance also the ability to avoid catastrophic situations or the need to satisfy a real-time response are considered as important system requirements[1]. We argue that the set of attributes presented in this paper can be used as a tool for agents adaptation to the environment in order to satisfy the desired goals of a stakeholder. There are two ways in which a stakeholder can take benefit of having these attributes in agents' definition. First, they can be aggregated to create *agents types*. For example, suppose we characterise a *social agent* as an agent who always communicates his goals to others when having a high intensity (he has defined the **Intensity*** attribute presented in the table 6). If we add to this social agent the **Insistence** attribute, we have a social agent who selects carefully his goals. Second, stakeholders can tune one or more of these attributes to get agents' behaviour more close to their needs.

As a long term goal we propose to create multi-agent systems in which a set of different agents types are tested under different simulated scenarios. The best performing sets could be selected to be used later in a real-time environments. The selection among different sets could then, be made automatically, changing a team of agents in different scenarios.

The role of attributes in a cognitive agents' architecture have been discussed by Sloman [15][16]. Following Correa[7][8], in our previous work [5][3], the attributes are associated to the definition of a mental state and it is explained how these attributes can increase the plasticity of the agents' reasoning process.

The two key roles of the attributes used in the agents' architecture are the following:

- To permit to control how the agents use their resources.
- To provide different ways of satisfying goals in different contexts.

For example, the insistence is related to how many times an agent persist on a specific strategy to satisfy a goal, while intensity discriminates the cases in which agents can execute a specific

Insistence	An agent with a high insistence, persists in achieving a goal. If he fails, he refuses to satisfy that goal again for a period $\delta f(x)$, where $f(x)$ depends on the time at which the agent spent on trying to achieve that goal.
Intensity	An agent communicates preys' positions only if his intensity is above a threshold. The rationale behind this is the following: An agent communicates only when he knows the probability of achieving alone the goal is small.
Intensity*	The probability of an agent communicates preys' position increases with the increasing of the value of his intensity attribute. The rationale behind is the following: An agent communicates his goals to the another agents to increase his probability of success.

Table 3: Different definitions of agents attributes using contextual parameters.

action to satisfy a goal by linking its execution capability to their energy resources at disposal in the execution context. Emotions are often mentioned along with the issues of agents' adaptation to changing environments, and control use of their resources [12]. Those and another related topics are also relevant in our research. For example in [14], emotion is regarded as a mechanism capable of creating action tendencies, which is related to our notion of agents' types. In [9] it is suggested the use of affective control states in which the same type of attributes we used are applied to control goals selection embedded in goal motivations.

Finally, the flexible selection of strategies which deal with time pressure is discussed in [13], which is related we used it in our work, i.e., on-line selection of the best strategy with respect to resource consumption.

References

- [1] Victor Basili, Paolo Donzelli, and Sima Asgari. A unified model of dependability: Capturing dependability in context. *IEEE Software*, 21(6):19–25, 2004.
- [2] M. Benda, V. Jagannathan, and R. Dodhiawala. On optimal cooperation of knowledge sources. Technical report, Boeing Artificial Intelligence Center, Boeing Computer Services, 1985.
- [3] Jose Cascalho, Luis Antunes, and Helder Coelho. Toward a motivated bdi using attributes embedded in mental states. In *XI Conferencia de la Asociación Española para la Inteligencia Artificial (CAEPIA 2005)*, volume 2, pages 215–224, 2005.
- [4] Jose Cascalho, Luis Antunes, Milton Correa, and Helder Coelho. Characterising agents' behaviours: selecting goal strategies based on attributes. In Matthias Klusch, Micahel Rovatsos, and Terry Payne, editors, *Cooperative Information Agents X*, volume 4149, pages 402–415. Springer, 2006.
- [5] Jose Cascalho, Leonel Nobrega, Milton Correa, and Helder Coelho. Exploring the mechanisms behind a bdi-like architecture. In *Conceptual Modeling Simulation Conference*, pages 153–158, 2005.
- [6] C. Castelfranchi and R. Conte. *Cognitive and Social Action*. UCL Press, 1995.
- [7] M. Corrêa and H. Coelho. From mental states and architectures to agents' programming. In H. Coelho, editor, *Proceedings of the Sixth Iberoamerican Conference in Artificial Intelligence*, volume 1484 of *Lectures Notes in Artificial Intelligence*, pages 64–75. Springer-Verlag, 1998.
- [8] Milton Corrêa and Helder Coelho. Collective mental states in an extended mental states framework. In *International Conference on Collective Intentionality IV, Certosa di Pontignano*, pages 13–15, 2004.

- [9] D. Davis and S. C. Lewis. Affect and affordance: Architectures without emotion. In AAAI, editor, *AAAI Spring symposium*, 2004.
- [10] Thomas Haynes and Sandip Sen. Evolving behavioral strategies in predators and prey. In Sandip Sen, editor, *IJCAI-95 Workshop on Adaptation and Learning in Multiagent Systems*, pages 32–37, Montreal, Quebec, Canada, 20-25 1995. Morgan Kaufmann.
- [11] J. Kok and N. Vlassis. The pursuit domain package. Technical report, Informatics Institute, University of Amsterdam, The Netherlands, 2003.
- [12] Luís Morgado and Graça Gaspar. Emotion based adaptive reasoning for resource bounded agents. In *AAMAS '05: Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*, pages 921–928, New York, NY, USA, 2005. ACM Press.
- [13] Sanguk Noh and Piotr J. Gmytrasiewicz. Flexible multi-agent decision making under time pressure. *Systems, Man and Cybernetics, Part A, IEEE Transactions on*, 2005.
- [14] E. Oliveira and L. Sarmento. Emotional valence-based mechanisms and agent personality. 2002.
- [15] A. Sloman. *Motive Mechanisms Emotions*, M.A. Boden (ed), *The Philosophy of Artificial Intelligence*, pages 231–247. Oxford University Press, 1990.
- [16] A. Sloman. Varieties of affect and the cogaff architecture schema, 2001.
- [17] Angelo Susi, Anna Perini, and John Mylopoulos. The tropos metamodel and its use. *Informatika*, 29(4):401–408, 2005.