An Integrative Modelling Approach for Simulation and Analysis of Adaptive Agents

Tibor Bosse[1], Catholijn M. Jonker[2], and Jan Treur[1]
[1]Vrije Universiteit Amsterdam, Department of Artificial Intelligence
{tbosse, treur}@cs.vu.nl, http://www.cs.vu.nl/~{tbosse, treur}
[2]Radboud Universiteit Nijmegen, Nijmegen Institute of Cognition and Information
C.Jonker@nici.ru.nl, http://www.nici.ru.nl/~catholj

Abstract

To simulate adaptive agents with abilities matching those of their real-world counterparts, a natural approach is to incorporate adaptation mechanisms such as classical conditioning into agent-based simulation. Existing models for adaptation mechanisms are usually based on quantitative methods such as DST. In contrast, agent-based simulation is usually based on qualitative, logical languages. To bridge this gap, this paper puts forward an integrative approach to simulate and analyse the conditioning process of an adaptive agent, which integrates quantitative and qualitative aspects within one temporal specification language. The approach comprises (1) simulation of adaptation mechanisms in an executable language, (2) automated analysis of dynamic properties against simulation traces, and (3) verification of representation relations for internal agent states against simulation traces. Furthermore, the approach addresses the issue of realism of intermediate states in a simulated conditioning process.

1. Introduction

Agent-based modelling techniques are often used to model and simulate (natural or artificial) agent systems that have to deal with dynamic and uncertain environments. Therefore, an important challenge for the area of agent-based modelling is the notion of adaptivity. An example of a basic mechanism for adaptation that can be found in many organisms is classical conditioning [10]. In order to create agent-based simulations with adaptive abilities matching those of their biological counterparts, a natural approach is to integrate such adaptation mechanisms into agent-based simulation models, e.g., [1].

In the literature, adaptation mechanisms such as classical conditioning are usually described and analysed informally. If formalisation is used, this is often based on mathematical models using differential equations, e.g., Dynamical Systems Theory (DST) [11]. In contrast, agent-based simulation models traditionally make use of qualitative, logical languages, such as Golog [12]. Most of these languages are appropriate for expressing qualitative relations, but less suitable to work with more complex numerical structures as, for example, in differential equations. Therefore, integrating such mathematical models within the design of agent-based simulation models is difficult. To achieve this integration, it is necessary to bridge the gap between the quantitative nature of existing adaptation models and the type of languages typically used in agent-based simulation.

In the area of simulation, a formalised model is used to compute the simulation steps. Languages and software environments are available to support this modelling process. Validation of a model is usually not formally supported; it is considerded a different issue. Often validation is done informally, by hand (or eye), based on comparison of a simulation trace with an empirical trace. In addition, sometimes specific (e.g., statistical) techniques are used to support certain aspects of validation. Usually in the domain that is modelled, *global properties* that should hold for the behaviour of a simulation model can be identified. As the languages used to specify a simulation model are directed to *local properties* (the steps between successive states), such global properties cannot be formalised in these languages. To obtain more support, also for validation of a simulation model, it is needed to integrate the modelling of such global properties in a formal manner as well, so that their specification and automated checking on simulation traces also can be supported by the modelling environment.

Proceedings of the 39th Annual Simulation Symposium (ANSS'06)
0-7695-2559-8/06 $20.00 © 2006 IEEE

In accordance with the findings mentioned above, this paper introduces an approach for simulation and analysis of adaptive agent behaviour and underlying mechanisms that is integrative in two ways: (1) it combines in one modelling framework both *qualitative, logical* and *quantitative, numerical* aspects, and (2) it enables modelling dynamics both at a *local level* (internal mechanisms of the agent) and at a *global level* (externally observable agent behaviour, and representation relationships between internal and external states).

Modelling dynamics at a local level concerns expressing temporal relationships between pairs of successive states, such as described by direct causal relations, or, for example, by the basic steps within an *adaptation mechanism*. A difference or differential equation is an example of a local level specification of dynamics. From a local perspective, the dynamics of the actual underlying (e.g., neural) mechanisms that play a role in the real world can be investigated. Local level specifications are the basis for the computation steps for a simulation model.

From the global perspective, more complex relationships over time can be used to model dynamics for adaptive agents. For example, the dynamics of *observed adaptive agent behaviour* can be analysed, i.e., how during a history of (learning) experiences, the behaviour is changing. For example, the performance of actions depending on a stimulus in the present and a certain training history (series of training stimuli in the past) can be modelled. This can take the form of a temporal relationship (an *input-output correlation*) involving a longer time duration and several agent input and output states over time.

Besides input-output correlations, also from a global perspective *representation relations* for intermediate, internal agent states can be modelled. During modelling, for an internal state of the agent, often a modeller has in mind a certain representational content, i.e., how it relates to other concepts outside the agent. To take a simple example, it may be expected that the internal belief that a horse is nearby correlates to the actual presence of this horse. Such expected representation relations may be inspired by knowledge of how the agent's adaptation mechanism is realised in Nature. The approach includes ways to (formally) specify such representation relations and verify them against simulation traces, showing whether this representational content is in accordance with the agent model's internal dynamics. In addition, for an adaptive biological agent with known neural mechanisms, the modelling approach enables validation of the agent's internal states in the model against its corresponding internal (neural) states in the world.

As both the adaptation mechanism and the externally observable behaviour are modelled in the form of temporal relationships, within the modelling approach it is also possible to logically relate the dynamics of internal agent models involving (neural) adaptation mechanisms to the model for the dynamics of the externally observable adaptive behaviour. Such *interlevel relations* can be useful in debugging a model, but also in the analysis of the circumstances under which a model will function well and under which not.

If the actual underlying neural mechanisms are included in the analysis of adaptive behaviour, the sea hare *Aplysia* is an appropriate species to study, since its neural mechanisms have been well-investigated [4]. In this paper it is shown how the proposed modelling approach for adaptive agents can be used to simulate and analyse both *Aplysia*'s adaptive behaviour and the underlying neural mechanisms. In Section 2, the approach is briefly introduced. Section 3 introduces the *Aplysia* case study. In Section 4 the executable local dynamic properties describing basic mechanisms for the case study are presented; simulations on the basis of these local dynamic properties are discussed in Section 5. In Section 6 the interlevel relations between dynamic properties of the externally observable behaviour and the local properties describing the internal mechanisms are discussed. Section 7 addresses the formalisation of representation relations. Section 8 discusses how all dynamic properties have been checked against the simulation traces. Section 9 is a discussion.

2. Modelling Approach

To formally specify dynamic properties that express criteria for representational content from a temporal perspective, an expressive language is needed. Dynamics will be described in the next section as evolution of *states* over time. The notion of state as used here is characterised on the basis of an ontology defining a set of *state properties* that do or do not hold at a certain point in time. Examples of state properties are 'the agent is hungry', 'the agent observes rain', or 'the environmental temperature is 7° C'. Real value assignments to variables are also considered as possible state property descriptions. For example, in a quantitative modelling approach (such as [11]), based on variables x_1, x_2, x_3, x_4, that are related by differential equations over time, value assignments such as $x_1 \leftarrow 0.06$, $x_2 \leftarrow 1.84$, $x_3 \leftarrow 3.36$, $x_4 \leftarrow -0.27$ are considered state descriptions. State properties are described by ontologies that specify the concepts used.

Based on such state properties, *dynamic properties* can be formulated that relate a state at one point in time

Proceedings of the 39th Annual Simulation Symposium (ANSS'06)

to one or more states at other points in time. A simple example is the following dynamic property:

'at any point in time t1 if the agent observes rain at t1, then there exists a point in time t2 after t1 such that at t2 the agent has internal state property s'

Here, for example, s can be viewed as a sensory representation of the rain. To express such dynamic properties, and other, more sophisticated ones, the temporal trace language TTL is used [6]. Within this language, explicit references can be made to time points and traces. Here a fixed *time frame* T is assumed which is linearly ordered. Depending on the application, it may be continuous (e.g., the real numbers), or discrete (e.g., the set of integers or natural numbers or a finite initial segment of the natural numbers), or any other form, as long as it has a linear ordering. Moreover, a *trace or trajectory* over an ontology Ont is a time-indexed sequence of states over Ont. The sorted predicate logic temporal trace language TTL is built on atoms referring to, e.g., traces, time and state properties. For example, 'in the internal state of agent A in trace γ at time t property s holds' is formalised by state(γ, t, internal(A)) $\models$ s. Here $\models$ is a predicate symbol in the language, usually used in infix notation, which is comparable to the Holds-predicate in situation calculus. Dynamic properties are expressed by temporal statements built using the usual logical connectives and quantification (for example, over traces, time and state properties).

To be able to perform some simulation experiments, a simpler temporal language has been used to specify executable models in a declarative manner. This language (the *leads to* language [3]) enables to model direct temporal dependencies between two state properties in successive states. This executable format is defined as follows. Let α and β be state properties of the form 'conjunction of atoms or negations of atoms', and e, f, g, h non-negative real numbers. Then the notation $\alpha \twoheadrightarrow_{e, f, g, h} \beta$, means:

If state property α holds for a certain time interval with duration g, then after some delay (between e and f) state property β will hold for a certain time interval of length h.

For a precise definition of the *leads to* format in terms of the language TTL, see [6]. A specification of dynamic properties in *leads to* format has as advantages that it is executable and that it can often easily be depicted graphically.

3. The Aplysia Case Study

To illustrate the proposed approach for modelling and simulation of adaptive agents, it is applied in a case study. As the topic of the case study, the sea hare *Aplysia* was chosen. The motivation for this choice is two-fold. First, *Aplysia* is a clear example of an adaptive agent. Second, the internal neural mechanisms of *Aplysia* are relatively simple, and therefore well understood. This enables the modeller to (formally) describe *Aplysia*'s behaviour both from an *internal* perspective (i.e., at a *local level*, considering neural mechanisms of the agent) and from an *external* perspective (i.e., at a *global level*, considering externally observable agent behaviour). As a result, both *interlevel relations* (see Section 6) and *representation relations* (see Section 7) can be established between both types of descriptions. First, in Section 3.1, *Aplysia*'s behaviour is described from an external perspective. In Section 3.2, *Aplysia*'s behaviour is described from an internal perspective.

3.1. External Perspective

Aplysia is a sea hare that is often used to do experiments. It is able to learn on the basis of classical conditioning. In this section, a simplified description is given of this learning behaviour (viewed from an external perspective), based on [4], pp. 155-156.

Behaviour before learning phase
Initially the following behaviour is shown:

- a tail shock leads to a response (contraction)
- a light touch on its siphon is insufficient to trigger such a response

Learning phase
Now suppose the following experimental protocol is undertaken. In each trial the subject is touched lightly on its siphon and then, shocked on its tail (as a consequence it responds).

Behaviour after a learning phase
It turns out that after a number of trials (three in the current example) the behaviour has changed: the animal also responds (contracts) on a siphon touch.

Note that, to characterise behaviour, there is a difference between the *learned* behaviour (which is simply an *adapted* stimulus-response behaviour) and the *learning* behaviour, which is a form of *adaptive* behaviour, no stimulus-response behaviour. To specify such behaviours the following sensor and effector states are used: tail_shock, siphon_touch, contraction. In terms of these state properties the following global dynamic properties can be specified in *leads to* format:

GP1 tail_shock $\twoheadrightarrow_{e,f,g,h}$ contraction (always)
GP2 siphon_touch $\twoheadrightarrow_{e,f,g,h}$ contraction (after learning)

However the learning behaviour itself is not expressable in *leads to* format, but it is in TTL format:

GP3 at any point in time t,
if a siphon touch occurs
and at three different earlier time points t1, t2, t3,
a siphon touch occurred, directly followed by a tail shock
then it will contract

Formally:

$\forall\gamma\ \forall t\ \text{state}(\gamma, t) \models \text{siphon\_touch}\ \&$
$\exists t1, t2, t3, u1, u2, u3\ \ t1 < u1 < t2 < u2 < t3 < u3 < t\ \&$
$\text{state}(\gamma, t1) \models \text{siphon\_touch}\ \ \&\ \text{state}(\gamma, u1) \models \text{tail\_shock}\ \ \&$
$\text{state}(\gamma, t2) \models \text{siphon\_touch}\ \ \&\ \text{state}(\gamma, u2) \models \text{tail\_shock}\ \ \&$
$\text{state}(\gamma, t3) \models \text{siphon\_touch}\ \ \&\ \text{state}(\gamma, u3) \models \text{tail\_shock}$
$\Rightarrow\ \exists t' \geq t\ \ \text{state}(\gamma, t') \models \text{contraction}$

As can be seen, the temporal complexity of the learning behaviour specification is much higher than that of the learned behaviour.

3.2. Internal Perspective

This section describes *Aplysia*'s behaviour from an internal perspective. The internal neural mechanism for *Aplysia*'s conditioning is depicted in Figure 1; cf. [4].

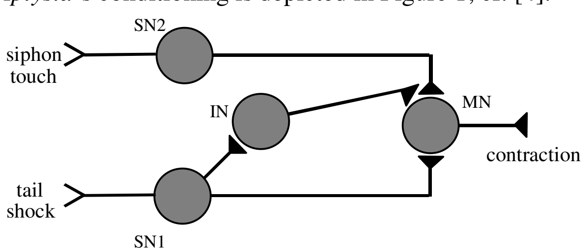

Figure 1. Neural mechanisms

A tail shock activates a sensory neuron SN1. Activation of this neuron SN1 activates the motoneuron MN; activation of MN makes the sea hare move. A siphon touch activates the sensory neuron SN2. Activation of this sensory neuron SN2 normally does not have sufficient impact on MN to activate MN. After learning, activation of SN2 has sufficient impact to activate MN. In addition, activation of SN1 also leads to activation of the intermediary neuron IN. If both SN2 and IN are activated simultaneously, this changes the synapse between SN2 and MN: it causes this synapse to produce more neurotransmitter if SN2 is activated. As a result, after a number of trials, activation of SN2 also yields activation of MN.

To model the example the following internal state properties are used:

SN1	sensory neuron 1 is activated
SN2	sensory neuron 2 is activated
IN	intermediary neuron IN is activated
MN	motoneuron MN is activated
S(r)	the synapse between SN2 and MN is able to produce an amount r of neurotransmitter

The dynamics of these internal state properties involve temporal *leads to* relationships, which are analysed in more detail in the next section.

4. Local Dynamic Properties

To model the internal dynamics of the example, the following local properties (in *leads to* format) are considered. They describe the basic parts of the process.

LP1 tail_shock $\rightarrow_{e,f,g,h}$ SN1
LP2 siphon_touch$\rightarrow_{e,f,g,h}$ SN2
LP3 SN1 $\rightarrow_{e,f,g,h}$ IN $\wedge$ MN
LP4 $S(r) \wedge SN2 \wedge IN \wedge r < 4 \rightarrow_{e,f,g,h} S(r+1)$
LP5 $S(4) \wedge SN2 \rightarrow_{e,f,g,h}$ MN
LP6 MN $\rightarrow_{e,f,g,h}$ contraction
LP7 $S(r) \wedge \text{not } S(r+1) \wedge r < 4 \rightarrow_{e,f,g,h} S(r)$
LP8 $S(4) \rightarrow_{e,f,g,h} S(4)$
LP9 start $\rightarrow_{e,f,g,h}$ S(1)

In Figure 2 an overview of these properties is given in a graphical form. Here, the circles denote state properties and the arrows denote dynamic properties.

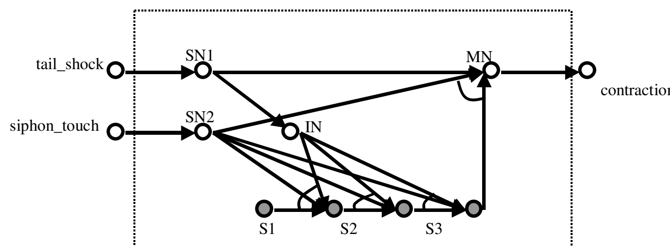

Figure 2. Overview of the basic dynamics of the simulation model

Note that this model is based on a number of simplifications. For example, it is assumed that after exactly 4 steps the strength of the synapse between SN2 and MN is maximal, and that there is no extinction. However, since our modelling approach supports the use of quantitative concepts (such as real numbers and mathematical operations), it is easy to incorporate such features in the model. A rather straightforward way to do this is by replacing LP4 through LP8 by the following local properties:

LP4 $S(r) \wedge SN2 \wedge IN \rightarrow_{e,f,g,h} S(\beta*(K-r)+(r*\varepsilon))$
LP5 $S(r) \wedge SN2 \wedge r > t \rightarrow_{e,f,g,h}$ MN
LP7 $S(r) \wedge \text{not } SN2 \rightarrow_{e,f,g,h} S(r*\varepsilon)$
LP8 $S(r) \wedge \text{not } IN \rightarrow_{e,f,g,h} S(r*\varepsilon)$

Here, β indicates the learning rate, K is the maximal strength of the synapse between SN2 and MN (e.g., 4), ε indicates the extinction rate, and t indicates the minimum threshold of S needed to have SN2 influence MN. For all values, real numbers can be used.

Another extension to the model is to introduce real-valued arguments for the state properties SN1, SN2, IN and MN as well, indicating the strength of their activation. This would allow the model to distinguish between, for example, tail shocks of different strengths. Although these extensions are relatively easy to perform, for reasons of presentation in the remainder of this paper the simplified model is used.

5. Simulation

As mentioned in the Introduction, local level specifications are the basis for the computation steps for a simulation model. Thus, special software environments can be created to enable the simulation of local level specifications, as long as these are in an executable format. For the executable language *leads to*, such a software environment has indeed been built, see [3] for details. Based on an input consisting of dynamic properties in *leads to* format, this software environment generates simulation traces. An example of such a trace can be seen in Figure 3. Here, time is on the horizontal axis, the state properties are on the vertical axis. A dark box on top of the line indicates that the property is true during that time period, and a lighter box below the line indicates that the property is false. This trace is based on all local properties identified in Section 4. In property LP1 and LP2 the values (0,0,1,3) have been chosen for the timing parameters e, f, g, and h. In all other properties, the values (0,0,1,1) have been chosen.

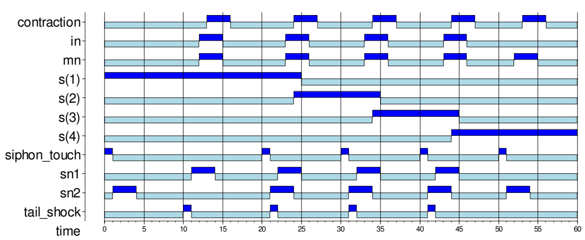

Figure 3. Example simulation trace

As can be seen in Figure 3, at the beginning of the trace the organism has not performed any conditioning. The initial siphon touch it receives does lead to the activation of sensory neuron SN2, but the synapse between SN2 and motoneuron MN does not produce much neurotransmitter yet (indicated by internal state property S(1)). Thus, the activation of SN2 does not yield an activation of MN, and consequently no external action follows. In contrast, it is shown that a shock of the organism's tail does initially lead to the external action of contraction. This can be seen in Figure 3 between time point 10 (when the tail shock occurs) and time point 13 (when the animal contracts). After that, the actual learning phase starts. This phase consists of a sequence of three trials where a siphon touch is immediately followed by a tail shock. As a result, the sensory neuron SN2 is activated at the same time as the intermediary neuron IN, which causes the synapse to change so that it can produce an increased amount of neurotransmitter each time SN2 is activated. Such a change in the synapse is indicated by a transition from one internal state property to another (first from S(1) to S(2), then to S(3), and finally to S(4)). As soon as internal state property S(4) holds (see time point 44), the conditioning process has been performed successfully. From that moment, *Aplysia*'s behaviour has changed: it also contracts on a siphon touch.

For the purposes of this example, the amount of trials is kept low (three). However, similar experiments have been performed with a case of 1000 learning steps. Since the abstract way of modelling used for the simulation is not computationally expensive, also these simulations took no more than 90 seconds. In addition, our simulation approach has possibilities to incorporate real numbers in state properties, and to perform complex mathematical operations with these numbers. This makes it more expressive than more traditional forms of temporal logic.

6. Interlevel Relations

In the previous sections, both the internal (neural) adaptation mechanism and the externally observable behaviour of *Aplysia* were modelled in the form of temporal relationships. Within the presented modelling approach, this implies that it is also possible to logically relate the dynamics of both models. This section outlines these *interlevel* connections between dynamic properties at different levels. It will be shown how the description at the level of the neurological mechanisms (the local dynamic properties LP1 through LP9) can be logically related to the description at the level of the overall behaviour (the global property GP3). This way, a formalisation is obtained of the (interlevel) reduction relation between the two levels. To be precise, this relation is described by the following implication:

(1) LP1 through LP9 & CWA $\Rightarrow$ GP3

This equation states that the local properties together imply the global property GP3 (which expresses that experiencing the combination of a tail shock and a siphon touch three times results in a response to the siphon touch alone). Moreover, one additional property is introduced, i.e., CWA. This second-order property that is commonly known as the Closed World Assumption expresses that at any point in time a state property that is not implied by a specification to be true is false. Let Th be the set of all local properties LP1 - LP9, then the formalisation is:

Closed World Assumption (CWA)
$\forall P \in \text{At(ONT)}\ \forall\gamma\ \forall t{:}\text{Th} \not\vdash \text{state}(\gamma,t) \models P \Rightarrow \text{state}(\gamma,t) \models \text{not } P$

The Closed World Assumption is needed to ensure that the intermediate results as indicated by the S(r) state properties can only hold as a result of the local properties LP1 through LP9, and not because of some other (mysterious) cause.

Essential milestones in the proof of relationship (1) are that subsequently S(1), S(2), S(3), and S(4) will hold. These milestones can be seen as the result of a learning process. Therefore, an additional lemma is introduced. This lemma describes the effect of a learning step on the synapse, showing the increase of parameter r in state property S(r), given that the siphon is touched, directly followed by a tail shock. In this case study the effect we are interested in is already reached at r=4. The lemma can easily be adapted for more lengthy learning processes. Formally, the lemma is specified as:

M(g, h, r) Learning step

$\forall\gamma\ \forall t1, t2, u1$
$t1 < u1 < t1 + g\ \&\ t1 < t2 < t1 + g\ \&\ r < 4\ \&$
$\forall t\ [t1 \leq t < t1 + h \Rightarrow state(\gamma, t) \models siphon\_touch\ ]\ \&$
$\forall t\ [u1 \leq t < u1 + h \Rightarrow state(\gamma, t) \models tail\_shock\ ]\ \&$
$\forall t\ [t2 \leq t < t2 + h \Rightarrow state(\gamma, t) \models S(r)\ ]$
$\Rightarrow\ \exists t3\ [\ t3 \geq t2\ \&\ \forall t\ [t3 \leq t < t3 + h \Rightarrow state(\gamma, t) \models S(r+1)\ ]\ ]$

Property M(g,h,r) can be proved for g=1, h=1, and r varying from 1 to 4 from LP1, LP2, LP3, LP4, LP7, and CWA, taking (0, 0, 1, 3) as timing parameters in LP1 and LP2, and (0, 0, 1, 1) for the timing parameters of the other local properties.

(2) LP1 & LP2 & LP3 & LP4 & LP7 & CWA $\Rightarrow$ M(1, 1, r)

The introduction of property M(1,1,r) allows one to reduce relationship (1) to the following, simpler implication:

(3) LP2 & LP5 & LP6 & LP7 & LP8 & CWA & M(1,1,r) $\Rightarrow$ GP3

The full proof of these interlevel relations is a difficult issue, and is left out of this paper. Nevertheless, the relations can be useful in the analysis of simulation traces. To illustrate this, assume that, in a given simulation trace, a certain global property (e.g., GP3) does not hold. Then by a refutation process it can be concluded that one of the lower level properties according to (3) does not hold either (i.e., LP2, LP5, LP6, LP7, LP8, LP9, CWA or M(1,1,r) does not hold). If, after checking these properties, it turns out that M(1,1,r) does not hold, then according to (2) either, LP1, LP2, LP3, LP4, LP7 or CWA does not hold. Thus, by this example refutation analysis eventually the cause of the unsatisfactory behaviour can be reduced to the failure of a local property.

7. Representational Content

In the literature on Philosophy of Mind different types of approaches to representational content of an internal state property have been put forward, for example the causal/correlational and relational specification approach; cf. [8], pp. 191-193, 200-202. These approaches to representational content have in common that the occurrence of the internal state property at a specific point in time is related to the occurrence of other state properties, at the same or at different time points. The 'other state properties' can be of two types: (A) *external world state properties*, independent of the agent, or (B) the agent's *interaction state properties* (i.e., sensor and effector properties). Furthermore, the type of relationships can be (1) purely functional *one-to-one correspondences*, (e.g., the correlational approach), or (2) they can involve more *complex relationships* with a number of states at different points in time in the past or future. So, four types of approaches to representational contents are distinguished, that can be indicated by codings such as A1, A2, and so on. Below, examples of such approaches are given.

According to the *causal/correlational approach* (see [8], pp. 191-193), the representational content of a certain internal state is given by a one-to-one correlation to another (in principle external) state property: type A1. For example, the internal belief that a horse is nearby is correlated to the actual presence of this horse, which is an external state property. Such an external state property may exist backward as well as forward in time. Hence, for the current example, in order to define the representational content of an internal state property, one should try if this can be related to a world state property that either existed in the past or will exist in the future. For example, the representational content for internal state property SN1 can be defined as world state property tail_shock, by looking backward in time. However, for some of the other internal state properties the representational content cannot be defined adequately according to the causal/correlational approach. In these cases, reference should not be made to one single state in the past or in the future, but to a temporal sequence of inputs or output state properties, which is not considered to adequately fit in the correlational approach. This shows that especially in cases where the agent learns from a number of trials extending over time, a classical approach to representational content is insufficient. Some authors even claim that it is a bad idea to aim for a notion of representation in such cases; e.g., [7,13].

As an alternative, the *Relational Specification approach* to representational content is based on a specification of how the occurrence of an internal state property relates to properties of (possibly many) other states distant in space and time; cf. [8], pp. 200-202. In this paper, for the 'other' states, interaction states are chosen: type B. The focus is on the B2 type, which is the more advanced case. Thus, the representational content of a certain internal state can be defined by specifying a temporal relation of the internal state property to sensor and action states in the past and future. An overview for the content of all internal state properties of the case study, according to the temporal relational specification approach is given, in an informal notation, in Table 1. Note that these relationships are defined at a semantic level. Different interaction state properties, separated by commas, should be read as the temporal sequence of these states.

Table 1. Relational Specification Approach (semantic level)

Internal State	Content (backward)	Content (forward)
S(2)	siphon_touch, tail_shock	
S(3)	siphon_touch, tail_shock, siphon_touch, tail_shock	
S(4)	siphon_touch, tail_shock, siphon_touch, tail_shock, siphon_touch, tail_shock	any siphon_touch is followed by contraction

Table 2 and 3 describe the same information as Table 1, but this time syntactically, expressed by TTL formulae. The following abstractions are used to describe training periods:

$\text{training\_up\_to}(\gamma,\ t1, u1, 1) \equiv\ u1 = t1 + 1\ \&$
$\text{state}(\gamma, t1) \models \text{siphon\_touch}\ \&\ \text{state}(\gamma, u1) \models \text{tail\_shock}$

$\text{training\_up\_to}(\gamma,\ t1, u2, 2) \equiv$
$\exists u1, t2\ [u1 < t2\ \&\ u2 = t2 + 1]$
$\text{training\_up\_to}(\gamma,\ t1, u1, 1)\ \&$
$\text{state}(\gamma, t2) \models \text{siphon\_touch}\ \&\ \text{state}(\gamma, u2) \models \text{tail\_shock}$

$\text{training\_up\_to}(\gamma,\ t1, u3, 3) \equiv$
$\exists u2, t3\ [u2 < t3\ \&\ u3 = t3 + 1]$
$\text{training\_up\_to}(\gamma,\ t1, u2, 2)\ \&$
$\text{state}(\gamma, t3) \models \text{siphon\_touch}\ \&\ \text{state}(\gamma, u3) \models \text{tail\_shock}$

Table 2. Relational Specification Approach (syntactic level, backward)

Int. st.	Content (backward)
S(2)	∀t1, u1 [training_up_to(γ, t1, u1, 1) & ¬∃t0 [training_up_to(γ, t0, u1, 2)] ⇒ ∃t2 > u1 [state(γ, t2) ⊨ S(2)]] ∀t1, u2 [training_up_to(γ, t1, u2, 2) & ¬∃t0 [training_up_to(γ, t0, u2, 3)] ⇒ ∃t3 > u2 [state(γ, t3) |=/= S(2)]]
S(3)	∀t1, u2 [training_up_to(γ, t1, u2, 2) & ¬∃t0 [training_up_to(γ, t0, u2, 3)] ⇒ ∃t3 > u1 [state(γ, t3) ⊨ S(3)]] ∀t1, u3 [training_up_to(γ, t1, u3, 3) & ¬∃t0 [training_up_to(γ, t0, u3, 4)] ⇒ ∃t4 > u3 [state(γ, t4) |=/= S(3)]]
S(4)	∀t1, u3 [training_up_to(γ, t1, u3, 3) & ¬∃t0 [training_up_to(γ, t0, u3, 4)] ⇒ ∃t4 > u3 [state(γ, t4) ⊨ S(4)]]

Consider, for example, the backward representational content of state property S(2). According to Table 2, the occurrence of exactly one learning trial (indicated by the fact that at u1, a training period up to 1 but not up to 2 has passed) eventually leads to a time point where S(2) holds. In addition, to make the content more precise, it is specified that the occurrence of exactly two learning trials eventually causes S(2) not to hold.

Table 3. Relational Specification Approach (syntactic level, forward)

Int. st.	Content (forward)
S(4)	∃ t' ≥ t [state(γ, t') ⊨ siphon_touch & ∀t' ≥ t [state(γ, t') ⊨ siphon_touch ⇒ ∃t" ≥ t' state(γ, t") ⊨ contraction]

As stated earlier, representational relations such as the ones specified here may correspond to certain expectations that the modeller has about the behaviour of the model. By (formally) specifying such expected representation relations and verifying them against simulation traces, it can be shown whether they are in accordance with the agent model's internal dynamics.

8. Checking Dynamic Properties

In addition to the simulation software, a software environment has been developed that enables to check dynamic properties specified in TTL against simulation traces. This environment takes a dynamic property and one or more (empirical or simulated) traces as input, and checks whether the dynamic property holds for the traces. It basically uses Prolog rules for the predicate sat that reduce the satisfaction of the temporal formula to the satisfaction of atomic state formulae at certain time points, which can be read from the trace representation. Using automatic checks of this kind, many of the properties presented in this paper have been checked against traces such as the one depicted in Figure 3. In particular, dynamic property GP3 (expressing the learning behaviour) has been checked successfully against all generated traces. Furthermore, the representation relations denoted in Table 2 have been checked. The duration of these checks varied from 1 to 3 seconds, depending on the complexity of the formula. They all turned out to be successful, which validates (for the given traces) our choice for the representational content of the internal state properties. However, note that these checks are only an empirical validation, they are no exhaustive proof as, e.g., model checking is. Currently, the possibilities are explored to combine TTL with existing model checking techniques.

9. Discussion

This paper introduces an integrative modelling approach for simulation and analysis of adaptive agent behaviour and underlying mechanisms. The approach is integrative in two ways. First, it combines both *qualitative, logical* and *quantitative, numerical* aspects in one modelling framework. Second, it enables modelling dynamics at a *local level* (internal neural mechanisms of the agent; cf. [5]) and at a *global level* (externally observable agent behaviour, and representation relationships between internal and external states).

The neural processes of the *Aplysia* case study (cf. [4]) have been formalised by identifying executable local dynamic properties for the basic dynamics of *Aplysia*'s neural conditioning mechanism. On the basis of these local properties simulations have been made. Moreover, it is shown how the description at the level of the neurological mechanisms can be logically related to the description of the overall behaviour, which can be considered as a formalisation of the (interlevel) reduction relations between the two levels. Such interlevel relations can be useful in the analysis of simulation traces, because they allow the modeller to reduce the failure of a global behavioural property to the failure of a local internal property of the model. This can be useful in debugging a model, but also in the analysis of the circumstances under which a model will function well and under which not.

Moreover, the presented approach allows the modeller to (formally) specify and check representation relations, which relate internal or intermediate states of the agent simulation model to other states of the model, possibly at different time points. In this paper, it was explored how representation relations can be defined for adaptive agents, using approaches such as in [10], pp. 200-202. The specifications of the representational content of the internal (neural) state properties for *Aplysia* have been validated by automatically checking them on the traces generated by the simulation model.

Finally, if the neural mechanisms of an adaptive biological agent are known, the modelling approach enables validation of the agent's internal states in the model against its corresponding internal (neural) states in the world. This way of validation can be applied in addition to the verification of representation relations, which is used to validate the dynamics of internal states of the agent simulation model against the observable behaviour of the agent. Thus, it can be verified to what extent the model satisfies *internal realism* in addition to external realism.

Concerning related work, in [2] another formal model is described of the dynamics of conditioning processes, using a similar modelling approach. However, that paper focuses on human conditioning, based on existing literature such as [9]. Instead, the current paper focuses on the specific case of *Aplysia*, of which the neural mechanisms are much simpler and therefore better understood. As a consequence, the model presented in the current paper is at a neural level, whereas the model of [2] is at a functional level. Another difference is that their model concentrates more on the temporal aspects of the conditioning.

References

[1] Balkenius, C. and Morén, J. Dynamics of a classical conditioning model. *Autonomous Robots, 7*, 1999, 41-56.

[2] Bosse, T., Jonker, C.M., Los, S.A., Torre, L. van der, and Treur, J. Formalisation and Analysis of the Temporal Dynamics of Conditioning. In: Mueller, J.P. and Zambonelli, F. (eds.), *Proc. of the 6th Int. Workshop on Agent-Oriented Software Engineering, AOSE'05*, 2005, pp. 157-168.

[3] Bosse, T., Jonker, C.M., Meij, L. van der, and Treur, J. LEADSTO: a Language and Environment for Analysis of Dynamics by SimulaTiOn. In: Eymann, T. et al. (eds.), *Proc. of the 3rd German Conf. on Multi-Agent System Technologies, MATES'05*. Springer LNAI 3550, 2005, pp. 165-178.

[4] Gleitman, H. *Psychology*, W.W. Norton & Company, New York, 1999.

[5] Hawkins, R.D., and Kandel, E.R. Is There a Cell-Biological Alphabet for Simple Forms of Learning? *Psychological Review*, vol. 91, 1984, pp. 375-391.

[6] Jonker, C.M., Treur, J., and Wijngaards, W.C.A. A Temporal Modelling Environment for Internally Grounded Beliefs, Desires and Intentions. *Cognitive Systems Research Journal*, vol. 4, 2003, pp. 191-210.

[7] Keijzer, F. Representation in Dynamical and Embodied Cognition. *Cognitive Systems Research Journal*, vol. 3, 2002, pp. 275-288.

[8] Kim, J., *Philosophy of Mind*. Westview Press, 1996.

[9] Machado, A. Learning the temporal Dynamics of Behaviour. *Psychological Review*, vol. 104, 1997, pp. 241-265.

[10] Pavlov, I.P. *Conditioned reflexes*. Oxford: Oxford University Press, 1927.

[11] Port, R.F. and van Gelder, T.J. *Mind as Motion: Explorations in the Dynamics of Cognition*. MIT Press, Cambridge, Mass., 1995.

[12] Reiter, R. *Knowledge in Action: Logical Foundations for Specifying and Implementing Dynamical Systems*. MIT Press, 2001.

[13] Sun, R. Symbol grounding: a new look at an old idea. *Philosophical Psychology*, vol.13, No.2, 2000, pp.149-172.

0-7695-2559-8/06 $20.00 © 2006 IEEE