Formalisation and Analysis of the Temporal Dynamics of Conditioning

Tibor Bosse[1], Catholijn M. Jonker[2], Sander A. Los[3], Leendert van der Torre[4,5] and Jan Treur[1]

[1]Vrije Universiteit Amsterdam, Department of Artificial Intelligence
De Boelelaan 1081a, NL-1081 HV Amsterdam, The Netherlands
[2]Radboud Universiteit Nijmegen, Nijmegen Institute for Cognition and Information, Montessorilaan 3, 6525 HR Nijmegen, The Netherlands
[3]Vrije Universiteit Amsterdam, Department of Cognitive Psychology,
Van der Boechorststraat 1, 1081 BT Amsterdam, The Netherlands
[4]Centrum voor Wiskunde en Informatica, Kruislaan 413, 1098 NL Amsterdam, The Netherlands
[5]Delft University of Technology, Julianalaan 134, 2628 BL Delft, The Netherlands

Abstract. In order to create adaptive Agent Systems with abilities matching those of their biological counterparts, a natural approach is to incorporate classical conditioning mechanisms into such systems. However, existing models for classical conditioning are usually based on differential equations. Since the design of Agent Systems is traditionally based on qualitative conceptual languages, these differential equations are often not directly appropriate to serve as an input for Agent System design. To deal with this problem, this paper explores a formal description and analysis of a conditioning process based on logical specification and analysis methods of dynamic properties of conditioning. Specific types of dynamic properties are global properties, describing properties of the process as a whole, or local properties, describing properties of basic steps in a conditioning process. If the latter type of properties are specified in an executable format, they provide a temporal declarative specification of a simulation model. Global properties can be checked automatically for simulated or other traces. Using these methods the properties of conditioning processes informally expressed by Los and Heuvel [7] have been formalised and verified against a specification of local properties based on Machado [8]'s mathematical model.

1 Introduction

Intelligent Agents often operate in dynamic and uncertain environments. Therefore, an important challenge for Agent-Oriented Software Engineering is to incorporate learning mechanisms into Agent Systems. A basic learning mechanism that can be found in many organisms is *classical conditioning*. Thus, in order to create Intelligent Agents Systems with abilities matching those of their biological counterparts, a natural approach is to build classical conditioning into such systems, e.g., [1].

However, in the literature classical conditioning is usually described and analysed informally. If formalisation is used, this is often based on mathematical models using differential equations, e.g., Dynamic Systems Theory [10]. In contrast, Agent-Based Systems traditionally make use of logical, conceptual languages, such as Golog [11] or 3APL [3]. Most of these languages are good for expressing qualitative relations, but less suitable to work with complex differential equations. Therefore, using mathematical models as a direct input for the design of Agent Systems is not trivial.

To bridge the gap between the quantitative nature of existing conditioning models and the conceptual, logical type of languages typically used to design Agent Systems, this paper introduces a logical approach for the analysis and formalisation of conditioning processes that combines qualitative and quantitative concepts, cf. [5]. Using this approach, the dynamics of conditioning are analysed both at a local and at a global level. First a local perspective model for temporal conditioning in a high-level executable format is presented, based on the idea of *local dynamic properties*. This executable model can be compared to (and was inspired by) Machado [8]'s differential equation model. Some simulation traces are shown. Next, as part of a non-local perspective analysis, a number of relevant *global dynamic properties* of the conditioning process are identified and formalised. These dynamic properties were obtained by formalising the informally expressed properties to characterise temporal conditioning processes, as put forward by Los and Heuvel [7]. It has been automatically verified that (under reasonable conditions) these global dynamic properties are satisfied by the simulation traces. Thus, it is validated that the local dynamic properties can be used as requirements for the design of adaptive agents. This finding offers possibilities to extend existing methodologies for Agent-Oriented Software Engineering by including learning mechanisms as observed in nature.

In Section 2, first some basic concepts of classical conditioning are introduced. Based on these concepts, Section 3 briefly describes Machado [8]'s mathematical model for conditioning. Next, Section 4 introduces our logical approach to modelling dynamic process, and Section 5 applies this approach to Machado's model. Some resulting simulation trace that were generated on the basis of the logical model are shown in Section 6. In Section 7, a number of relevant global dynamic properties are described (cf. [7]), that are expected to hold for conditioning processes. In Section 8 these global properties are automatically checked against the simulation model. Section 9 concludes the paper with a discussion.

2 Basic Concepts of Conditioning

Research into conditioning is aimed at revealing the principles that govern associative learning. To this end, several experimental procedures have been developed. In classical conditioning, an organism is presented with an initially neutral conditioned stimulus (e.g., a bell) followed by an unconditioned stimulus (e.g., meat powder) that elicits an innate or learned unconditioned response in the organism (e.g., saliva production for a dog). After acquisition, the organism elicits an adaptive conditioned response (also saliva production in the example) when the conditioned stimulus is presented alone. In operant conditioning, the production of a certain operant response that is part to the volitional repertoire of an organism (e.g., bar pressing for a rat) is strengthened after repeated reinforcement (e.g., food presentation) contingent on the operant response.

In their review, Gallistel and Gibbon [4] argued that these different forms of conditioning have a common foundation in the adaptive timing of the conditioned (or operant) response to the appearance of the unconditioned stimulus (or reinforcement). This feature is most apparent in an experimental procedure called trace conditioning, in which a blank interval (or 'trace') of a certain duration separates the conditioned and unconditioned stimulus (in classical conditioning) or subsequent reinforcement phases (in operant conditioning). In either case, the conditioned (or operant) response obtains its maximal strength, here called *peak level*, at a moment in time, called *peak time*, that closely corresponds to the moment the unconditioned stimulus (or reinforcement) occurs.

For present purposes, we adopt the terminology of an experimental procedure that is often used to study adaptive timing and the possible role of conditioning in humans. In this procedure, a trial starts with the presentation of a *warning stimulus* (S1; comparable to a conditioned stimulus). After a blank interval, called the *foreperiod* (FP), an *imperative stimulus* (S2, comparable to an unconditioned stimulus) is presented to which the participant responds as fast as possible. The *reaction time* (RT) to S2 is used as an estimate of the conditioned state of preparation at the moment S2 is presented.

In this type of research, FP is usually varied at several discrete levels. That is, S2 can be presented at several moments since the offset of S1, which are called *critical moments*. The moment that is used for the presentation of S2 on any given trial is called the *imperative moment* of that trial. In a *pure block*, the same FP is used across all trials of that block. That is, in a pure block there is one critical moment that corresponds to the imperative moment on each trial. In a *mixed block*, all levels of FP occur randomly across trials. That is, a mixed block has several critical moments, but on any specific trial, only one of the moments is the imperative moment.

3 Modelling by Differential Equations

Machado [8] presented a basic model of the dynamics of a conditioning process. The structure of this model, with an adjusted terminology as used by [6], is shown in Figure 1. The model posits a layer of *timing nodes* (Machado calls these *behavioral states*) and a single *preparation node* (called *operant response* by Machado). Each timing node is connected both to the next timing node and to the preparation node. The connection between each timing node and the preparation node (called *associative link* both by Machado and within the current paper) has an adjustable weight associated to it. Upon the presentation of a warning stimulus, a cascade of activation propagates through the timing nodes according to a regular pattern. Owing to this regularity, the timing nodes can be likened to an internal clock or pacemaker. At any moment, each timing node contributes to the activation of the preparation node in accordance with its activation and its corresponding weight. The activation of the preparation node reflects the subject's preparatory state, and is as such related to reaction time for any given imperative moment.

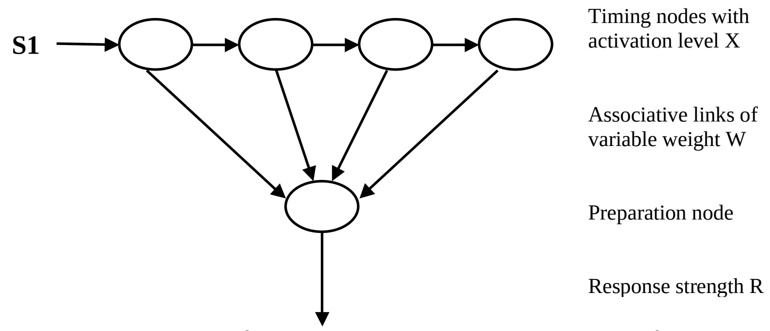

Fig. 1. Structure of Machado's conditioning model (adjusted from [8])

The weights reflect the state of conditioning, and are adjusted by learning rules, of which the main principles are as follows. First, *during* the foreperiod extinction takes place, which involves the decrease of weights in real time in proportion to the activation of their corresponding timing nodes. Second, *after* the presentation of the imperative stimulus a process of reinforcement takes over, which involves an increase of the weights in accordance with the current activation of their timing nodes, to preserve the importance of the imperative moment. In [8] the more detailed dynamics of the process are given by a mathematical model (based on linear differential equations), representing the (local) temporal relationships between the variables involved. For example, $d/dt\ X(t,n) = \lambda X(t,n-1) - \lambda X(t,n)$ expresses how the activation level of the n-th timing node $X(t+dt,n)$ at time point $t+dt$ relates to this level $X(t,n)$ at time point t and the activation level $X(t,n-1)$ of the $(n-1)$-th timing node at time point t.

4 Modelling by Dynamic Properties

As discussed above, mathematical models based on differential equations can be used to model local temporal relationships within conditioning processes. However, conditioning processes can also be characterised by temporal relationships of a less local form. As an example, taken from [7], a dynamic property can be formulated expressing the monotonicity property that 'the response level increases before the critical moment is reached and decreases after this moment'. This is a more global property, relating response levels at any two points in time before the critical moment (or after the critical moment). Therefore it is useful to explore formalisation techniques, as an alternative to differential equations, to express not only for local properties, but also for non-local properties. A second limitation of differential equations is that they are based on quantitative (calculational) relationships, whereas also non-quantitative aspects may play a role (for example, the monotonicity property mentioned above). This suggests that it may be useful to explore alternative formalisation techniques for dynamic properties of conditioning processes that allow one to express both quantitative and non-quantitative aspects.

As already mentioned in the Introduction, the approach presented in this paper indeed uses alternative formalisation languages to express dynamic properties of conditioning processes, both for local and global properties and both for quantitative and non-quantitative aspects. To this end the *Temporal Trace Language* TTL is used as a tool. For a detailed introduction to this language, see [5]. For an example of a previous application to the simulation and analysis of cognitive processes, see [2]. In the next sections TTL will be used to describe dynamic properties of a conditioning process at different levels of aggregation. At the lowest level of aggregation, *local dynamic properties* are dynamic properties of the basic mechanisms of the conditioning process. Since these properties are executable, they can (and will) be used to create a simulation model of a conditioning process (comparable to and inspired by Machado’s model). At a higher level of aggregation, *global dynamic properties*, i.e., properties of the conditioning process as a whole, will be expressed (e.g., indicating how a certain pattern of behaviour has been changed by a conditioning process). These dynamic properties were obtained by formalising the informally expressed properties to characterise temporal conditioning processes, as put forward by Los and Heuvel [7]. In addition, it will be shown that the global properties are satisfied by the traces generated on the basis of the local properties.

5 Local Dynamic Properties

A selection of the local properties (LPs) we defined in order to describe the basic mechanisms of the conditioning process is presented below. A local property generally has the format $\alpha \twoheadrightarrow \beta$, indicating that α leads to β, after a certain (specified) delay. The concepts used within the dynamic properties (called *state properties*) are described in Table 1.

Table 1. State Properties

X(n,u)	Timing node n has activation level u. In the current simulation, n ranges over the discrete domain [0,5]. Thus, our model consists of six timing nodes. The activation level u can take any continuous value in the domain [0,1].
W(n,v)	Associative link n has weight v. Again, n ranges over the discrete domain [0,5]. The weight v can take any continuous value in the domain [0,1].
R(r)	The preparation node has response strength r (a continuous value in the domain [0,1]).
S1(s)	Warning stimulus S1 occurs with strength s. Within our example, s only takes the values 0.0 and 1.0. However, the model could be extended by allowing any continuous value in-between.
S2(s)	Imperative stimulus S2 occurs with strength s.
Xcopy(n,u)	Timing node n had activation level u at the moment of the occurrence of the last imperative stimulus (S2). See dynamic property LP4 and LP6.
instage(ext)	The process is in a stage of extinction. This stage lasts from the occurrence of S1 until the occurrence of S2.
instage(reinf)	The process is in a stage of reinforcement. This stage starts with the occurrence of S2, and lasts during a predefined reinforcement period (e.g. 150 msec).
instage(pers)	The process is in a stage of persistence. This stage starts right after the reinforcement stage, and lasts until the next occurrence of S1.

As Machado [8]’s model was used as a source of inspiration, for some of the properties presented below the comparable differential equation within Machado's model is given as well. However, since Machado's mathematical approach differs at several points from the logical approach presented in this paper, there is not always a straightforward 1:1 mapping between both formalisations. For instance, state property X(n,u) within our TTL formalisation has a slightly different meaning than the corresponding term X(t,n) in Machado's differential equations. In the former, n stands for the timing node, u stands for the activation level, and X(n,u) stands for the fact that timing node n has activation level u. In the latter, t stands for a time point, n for the timing node, and X(t,n) as a whole for the activation level.

Using the concepts described in Table 1, the following local properties have been specified to describe the basic mechanisms of the conditioning process:

LP1 Initialisation. The first local property LP1 expresses the initialisation of the values for the timing nodes and the associative links. Formalisation (for n ranging over [0,5]):

start $\twoheadrightarrow$ X(n, 0) $\wedge$ W(n, 0)

LP2 Activation of initial timing nodes. Local property LP2 expresses the activation (and adaptation) of the 0^{th} timing node. Immediately after the occurrence of the warning stimulus (S1), this state has full strength. After that, its value decreases until the next warning stimulus. Together with LP3, this property causes the spread of activation across the timing nodes. Here, $\lambda > 0$ is a rate parameter that controls the speed of this spread of activation, and step is a constant indicating the smallest time step in the simulation. For the simulation experiments presented in the next section, λ was set to 10 and step was set to 0.05.

X(0, u) $\wedge$ S1(s) $\twoheadrightarrow$ X(0, u*(1-λ*step)+s)

Comparable differential equation in Machado [8]'s model:

d/dt X(t,0) = -λX(t,0).

LP3 Adaptation of timing nodes. LP3 expresses the adaptation of the n^{th} timing node (for n ranging over [1,5]), based on its own previous state and the previous state of the $n\text{-}1^{th}$ timing node. Together with LP2, this property causes the spread of activation across the timing nodes. Here, λ is a rate parameter that controls the speed of this spread of activation (see LP2).

X(n, u1) $\wedge$ X(n-1, u0) $\twoheadrightarrow$ X(n, u1+λ*(u0-u1)*step)

Comparable differential equation in Machado [8]'s model:

d/dt X(t,n) = λX(t,n-1) - λX(t,n).

LP4 Storage of timing nodes at moment of reinforcer. LP4 is needed to store the value of the n^{th} timing node at the moment of the occurrence of the imperative stimulus (S2). These values are used later on by property LP6.

X(n, u) $\wedge$ S2(1.0) $\twoheadrightarrow$ Xcopy(n, u)

LP5 Extinction of associative links. LP5 expresses the adaptation of the associative links during extinction, based on their own previous state and the previous state of the corresponding timing node. Here, α is a learning rate parameter. For the simulation experiments presented in the next section, the value 2 was chosen for α, inspired by [6]. This rather high value for α causes the model to adjust quickly to changing temporal regimes.

instage(ext) $\wedge$ X(n, u) $\wedge$ W(n, v) $\twoheadrightarrow$ W(n, v*(1-α*u*step))

Comparable differential equation in Machado [8]'s model:

d/dt W(t,n) = -αX(t,n)W(t,n)

LP6 Reinforcement of associative links. LP6 expresses the adaptation of the associative links during reinforcement, based on their own previous state and the previous state of Xcopy. Here, β is a learning rate parameter. For the simulation experiments presented in the next section, the value 2 was chosen for β, inspired by [6].

instage(reinf) $\wedge$ Xcopy(n, u) $\wedge$ W(n, v) $\twoheadrightarrow$ W(n, v*(1-β*u*step) + β*u*step)

Comparable differential equation in Machado [8]'s model:

d/dt W(t,n) = βX(T,n)[K-W(t,n)].

LP7 Persistence of associative links. LP7 expresses the persistence of the associative links at the moments that there is neither extinction not reinforcement.

instage(pers) ∧ W(n, v) ↠ W(n, v)

LP8 Response function. LP8 calculates the response by adding the discriminative function of all states (i.e., their associative links * the degree of activation of the corresponding state).

W(1, v1) ∧ W(2, v2) ∧ W(3, v3) ∧ W(4, v4) ∧ W(5, v5) ∧ X(1, u1) ∧ X(2, u2) ∧ X(3, u3) ∧ X(4, u4) ∧ X(5, u5) ↠ R(v1*u1 + v2*u2 + v3*u3 + v4*u4 + v5*u5)

Note that the translation from differential equations to local properties in TTL is relatively easy to make. Assuming some experience with both kinds of modelling, a set of differential equations as given above can be translated within a couple of hours.

6 Simulation Examples

A software environment has been developed that generates simulation traces of the conditioning process, based on an input consisting of dynamic properties in formal format. A large number (about 20) of such traces have been generated, with different parameters for foreperiod (50, 100, 150, 200, 300, and 500 msec) and block type (pure blocks and mixed blocks), selected on the basis of [6]. An example of such a trace can be seen in Figure 2. Here, time is on the horizontal axis. Each time unit corresponds to 50 msec. The relevant concepts (S1, instage(ext), instage(pers) and R) are on the vertical axis. This trace describes the dynamics of a person that has been subject to conditioning in a pure block with a foreperiod of 6 time units, i.e., 300 msec. As can be seen in the trace, the subject is maximally prepared (response strength r > 0.4) at about 6 time units (i.e., 300 msec) after the occurrence of the warning stimulus (S1(1.0)).

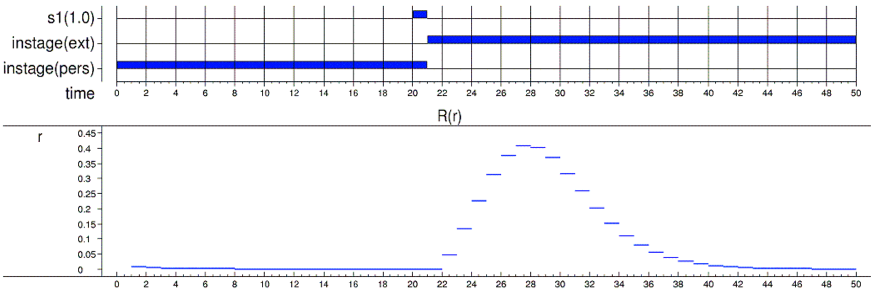

Fig. 2. Example simulation trace (pure block, FP=300 msec)

Figure 3 shows the case of a mixed block. In this situation, two types of foreperiod (FP=100 and FP=500 msec) have randomly been presented during the preceding trials. As a consequence, the curves that plot the response level have two peaks: one for each critical moment. The current trace shows two trials: one in which the imperative moment corresponds to the first critical moment, and one in which it corresponds to the second critical moment.

As mentioned above, a number of similar experiments have been performed, with different parameters for foreperiod and block type. The results were consistent with the data produced by Machado.

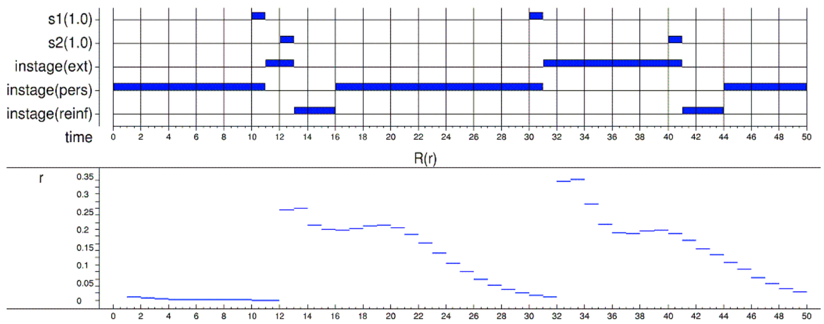

Fig. 3. Example simulation trace (mixed block, FP=100 & 500 msec)

7 Analysis of Global Dynamic Properties

In [7], the following properties of the overall conditioning process are put forward:

'Corresponding to each critical moment there is a state of conditioning, the adjustment of which is governed by learning rules of trace conditioning (specified subsequently).'

(1) 'The state of conditioning implicates an increase and decay of response-related activation as a critical moment is bypassed in time.'

(2) 'the conditioned response takes more time to build up and decay and its corresponding asymptotic value is lower when its corresponding critical moment is more remote from the warning signal.'

(3) 'on any trial, the strength of the conditioned response corresponding to a critical moment is reinforced (i.e., increased toward its asymptote) if and only if that critical moment coincides with the imperative moment.'

(4) 'on any trial the strength of the conditioned response is extinguished (i.e., driven away from its asymptote) if and only if its corresponding critical moment occurs before the imperative moment, whereas it is left unaffected if its corresponding critical moment occurs later than the imperative moment.' [7], p. 372.

These properties have a rather informal and non-mathematical nature. Below it is shown how they can be denoted in semi-formal notation. Because of space limitations, only the first property will also be presented in formal notation (TTL).

GP1 **has_global_hill_prep(γ,t1,t2,s1,a,u)**

The first global property GP1 is a formalisation of informal property (1) presented above. It describes the following: In trace γ, if at t1 a stimulus s1 starts, then the preparation level for action a will increase from t1 until t2 and decrease from t2 until t1 + u, under the assumption that no stimulus occurs too soon (within u time) after t1. Formally:

∀t', t", s', p', p", x, x'
stimulus_starts_at(γ, t1, s1, x) & ¬ stimulus_starts_within(γ, t1, t1+u, s', x') &
has_preparation_level_at(γ, t', p', a) & has_preparation_level_at(γ, t", p", a)
⇒ [t1 ≤ t' < t" ≤ t2 & t" ≤ t1 + u ⇒ p' < p"] &
[t2 ≤ t' < t" ≤ t1 + u ⇒ p' > p"]

GP2 **pending_peak_versus_critical_moment(γ1,γ2,t1,t2,c1,c2)**

Global property GP2 is a formalisation of informal property (2). It describes that: If for trace γ2 at time t2 peak time c2 is more remote than peak time c1 for γ1 at time t1, then at t2 in γ2 the pending peak level is lower than the pending peak level at t1 in γ1.

GP3 **dynamics_of_pending_preparation(γ,t1,t2,c,v,p,p',s1,s2,a, d,ε)**

GP3 is a formalisation of both informal property (3) and (4) together. It describes the following:

If t1 < t2
and at t1 the pending preparation level for time t1+v, action a, and stimuli s1 and s2 is p,
and at t2+d the pending preparation level for time t2+d+v, action a, and stimuli s1 and s2 is p',
and in trace γ at time t1 a stimulus s1 starts,
and in trace γ at time t2 a stimulus s2 starts,
and in trace γ the maximum peak level for a is pmax,
and in trace γ the minimum preparation level for a is pmin,
then:

t2 ∈ [t1+c-ε, t1+c+ε]	iff p' > p	(reinforcement, given p<pmax)
t2 > t1+c+ε	iff p' < p	(extinction, given that p > pmin)
t2 < t1+c-ε	iff p' = p	(persistence)

Parameter d refers to the time needed to process the events (d > 0), and c refers to a critical moment.

8 Checking Global Properties on Traces

In addition to the software described in the Simulation section, other software has been developed that takes traces and formally specified properties as input and checks whether a property holds for a trace. Using automatic checks of this kind, the four formalised properties based on [7] have been checked against traces like the ones depicted in Figure 2 and 3. This section discusses the results of these checks.

GP1 **has_global_hill_prep(γ,t1,t2,s1,a,u)**

This property turned out to hold for the generated traces, as long as reasonable values are chosen for the parameters. In particular, the parameters should meet the following conditions:

- t1 = a time point when s1 occurs
- t2 = t1 + duration of s1 + length of FP
- u = iti (the intertrial interval during the preceding conditioning process)

For example, the following property holds: has_global_hill_prep(γ1, 20, 27, s1, a, 20), where γ1 is the trace provided in Figure 2. Thus, for this trace the following holds: if at time point 20 a stimulus s1 starts, then the preparation level for action a increases from 20 until 27 and decreases from 27 until 40, under the assumption that no stimulus occurs between 20 and 40.

GP2 **pending_peak_versus_critical_moment(γ1,γ2,t1,t2,c1,c2)**

Checking property GP2 involves comparing two traces. Basically, it states that in traces where the foreperiod is longer, the level of response is lower. In order to check GP2, several traces have been generated that are similar to the trace in Figure 2, but each with a different foreperiod. For all combinations of traces, the property turned out to hold. To take an example, the following property holds:

pending_peak_versus_critical_moment(γ1, γ2, 20, 20, 6, 7), where γ1 is the trace provided in Figure 2, and γ2 is a similar trace with FP=7. This means that, if for trace γ2 at time 20 peak time 7 is more remote than peak time 6 for γ1 at time 20, then at 20 in γ2 the pending peak level is lower than the pending peak level at 20 in γ1.

GP3 dynamics_of_pending_preparation(γ,t1,t2,c,v,p,p',s1,s2,a, d,ε)
Property GP3 combines property (3) and (4) as mentioned in the previous section. Basically, the property consists of three separate statements that relate the strength of the conditioned response (p) to the critical moment (t1+c) and the imperative moment (t2), by stating that:

GP3A. p increases iff t2 = t1+c
GP3B. p decreases iff t2 > t1+c
GP3C. p remains the same iff t2 < t1+c

An example of this property with reasonable parameter values is: dynamics_of_pending_preparation(γ, 10, 12, 10, 10, p, p', s1, s2, a, 18, 0), where γ is the trace depicted in Figure 3. However, this property turned out not to hold. A close examination of Figure 3 will reveal the cause of this failure. This trace describes a mixed block with two types of foreperiod (FP=2 and FP=10). At time point 10, a warning stimulus (S1) occurs. At this time point, the pending preparation level for the latest critical moment (time point 20) has a certain value. And since this critical moment occurs after the occurrence of S2 (the imperative moment: time point 12), the pending preparation level for the latest critical moment should remain the same, according to property GP3C above. However, in the trace in question this is not the case (see Figure 3: in the second curve the second peak is slightly lower than in the first curve). Hence, it may be concluded that property GP3C (sub-property persistence presented earlier) does not hold for the chosen parameters. Fortunately, an explanation of this finding can be found in a later section of [7], where the authors revise their original model as follows:

'According to the original model, extinction and reinforcement affect each state of conditioning in an all-or-none way, thereby excluding a coupling between states of conditioning corresponding to adjacent critical moments. According to the revised model, extinction and reinforcement affect the states of conditioning more gradually across the time scale, resulting in a coupling between adjacent states.' [7], p. 383.

The revision of the model also implies a revision of property GP3. To be more specific, sub-property persistence can be changed into the following:

$t2 < t1 + c - \varepsilon$ iff $p' \in [\, p - \delta, p + \delta \,]$

Here, δ is a tolerance factor allowing a small deviation from the strength of the original response. After adapting GP3 accordingly, the property turned out to hold.

9 Discussion

To bridge the gap between the quantitative nature of existing conditioning models and the conceptual, logical type of languages typically used to design Agent Systems, in this paper a logical approach was introduced for the analysis and formalisation of conditioning processes that combines qualitative and quantitative concepts. Using this approach, the dynamics of conditioning have been analysed both at a local and at a global level.

From a local perspective, a model for temporal conditioning in a high-level executable format was presented, based on the idea of local dynamic properties. This model can be compared to (and was inspired by) Machado [8]'s differential equation model, and has been used to generate a number of simulation traces.

Next, as part of a non-local perspective analysis, a number of relevant *global dynamic properties* of the conditioning process have been identified and formalised. It has been confirmed, by means of formal verification, that the assumptions of the informal conditioning model proposed by Los and Heuvel [7] are global properties of the formal model developed by Machado [8], given certain restrictions of the parameter values, and given slight adaptations of the persistence rule given by GP3C. This is an important finding, because the global properties have proved to be highly useful in accounting for key findings in human timing, see [6], [7]. Thus, it was validated that the local dynamic properties can be used as requirements for the design of adaptive agents. As a result, existing methodologies for Agent-Oriented Software Engineering can be extended by including learning mechanisms as observed in nature. Currently, most research on reinforcement learning in Multi-Agent Systems concentrates only on the correctness of a response, not on its timing. By considering temporal aspects, the research presented in this paper is novel.

One crucial finding the global properties can deal with effectively is the occurrence of sequential effects of FP. These effects entail that on any given trial, RT is longer when the FP of that trial is shorter than the FP of the preceding trial relative to when it is as long as or longer than the FP of the preceding trial. Stated differently, RT is longer when the imperative moment was bypassed during the FP on the preceding trial than when it was not bypassed during FP on the preceding trial, see, e.g., [7], [9]. This finding is well accounted for by the learning rules formulated as GP3. According to GP3B, the state of conditioning (p) associated with a critical moment is subject to extinction when a critical moment is bypassed during FP (i.e., $t_2 > t_1 + c$), which is neither the case for the imperative moment, where according to GP3A reinforcement occurs (i.e., $t_2 > t_1 + c$), nor for critical moment beyond the imperative moment, where the state of conditioning persists according to GP3C (i.e., $t_2 < t_1 + c$). Note that the adjustment of GP3C suggested by the present check of global properties does not compromise the effectiveness of these learning rules, because the tolerance factor δ is small relative to the extinction described by GP3B.

In fact, the addition of the tolerance area δ to GP3C, might prove to be helpful in accounting for a more subtle effect in the extant literature. This concerns the finding that the FP-RT functions obtained in pure and mixed blocks cross over at the latest critical moment. Specifically, in pure blocks, the FP-RT function has been found to be upward sloping, given a minimal FP of about 250 – 300 msec. By contrast, in mixed blocks, the RT is slowest at the shortest critical moment (due to the influence of sequential effects described in the previous paragraph) and decreases as a negatively accelerating function of FP. At the latest critical moment the pure and mixed FP-RT functions come together, presumably because this moment is never bypassed during FP on the preceding trial, allowing the state of conditioning to approach its asymptotic value in either case. Sometimes, though, a cross-over of the two FP-RT functions is reported, which has been shown to be particularly pronounced in certain clinical populations, such as people diagnosed with schizophrenia (see [12] for a review). This finding may be related to the failure to confirm GP3C without the allowance of a tolerance area δ. Thus, it could be that, for certain parameter settings, the state of conditioning corresponding to the latest critical moment approaches its asymptotic value more closely when a shorter FP occurred on the preceding trial (which is often the case in mixed blocks) than when the same FP occurred on the preceding trial (as is always the case in pure blocks).

Since the results of our simulation model were found to be consistent with the model of [8], our model was implicitly compared with empirical work. However, in future work, it will be compared explicitly with empirical data. Since the checking software can take traces of different format as input, it will be possible to verify the global properties shown in Section 7 against experimental human conditioning traces.

Acknowledgements

This paper was improved thanks to some valuable comments of anonymous referees.

References

1. Balkenius, C. and Morén, J. (1999). Dynamics of a classical conditioning model. *Autonomous Robots, 7*, 41-56.
2. Bosse, T., Delfos, M.F., Jonker, C.M., and Treur, J. (2003). Analysis of Adaptive Dynamical Systems for Eating Regulation Disorders. *Proc. of the 25th Annual Conf. of the Cognitive Science Society, CogSci'03*. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.
3. Dastani, M., Dignum, F., and Meyer, J-J.Ch. (2003). *3APL: A Programming Language for Cognitive Agents*. ERCIM News, European Research Consortium for Informatics and Mathematics, Special issue on Cognitive Systems, No. 53.
4. Gallistel, C.R. and Gibbon, J. (2000). Time, rate, and conditioning. *Psychological Review*, vol. 107, pp. 289-344.
5. Jonker, C.M., Treur, J., and Wijngaards, W.C.A. (2003). A Temporal Modelling Environment for Internally Grounded Beliefs, Desires and Intentions. *Cognitive Systems Research Journal,* vol. 4(3), pp. 191-210.
6. Los, S.A., Knol, D.L., and Boers, R.M. (2001). The Foreperiod Effect Revisited: Conditioning as a Basis for Nonspecific Preparation. *Acta Psychologica*, 106, pp. 121-145.
7. Los, S.A. and van den Heuvel, C.E. (2001). Intentional and Unintentional Contributions to Nonspecific Preparation During Reaction Time Foreperiods. *Journal of Experimental Psychology: Human Perception and Performance*, vol. 27, pp. 370-386.
8. Machado, A. (1997). Learning the temporal Dynamics of Behaviour. *Psychological Review*, vol. 104, pp. 241-265.
9. Niemi, P. and Naatanen, R. (1981). Foreperiod and Simple Reaction Time. *Psychological Bulletin*, vol 89, pp. 133-162.
10. Port, R.F. and van Gelder, T.J. (1995). *Mind as Motion: Explorations in the Dynamics of Cognition*. MIT Press, Cambridge, Mass.
11. Reiter, R. (2001). *Knowledge in Action: Logical Foundations for Specifying and Implementing Dynamical Systems*. MIT Press.
12. Rist, F. and Cohen, R. (1991). Sequential effects in the reaction times of schizophrenics: crossover and modality shift effects. In E.R. Steinhauer, J.H. Gruzelier, & J. Zubin, *Handbook of schizophrenia*, vol. 5: Neuropsychology, psychophysiology and information processing (pp. 241-271). Amsterdam: Elsevier.