An Information Theory Based Anticipation Architecture

Philippe Capdepuy*, Daniel Polani*†, and Chrystopher L. Nehaniv*†

Adaptive Systems* and Algorithms† Research Groups
School of Computer Science, University of Hertfordshire
College Lane, Hatfield, Herts, AL10 9AB, UK
{P.Capdepuy,D.Polani,C.L.Nehaniv}@herts.ac.uk

Abstract. We introduce information theoretic tools that can be used in an animat for constructing an internal predictive model. This model is based on two different kinds of causal relationships: time-delay relationship, where two events are related by a nearly constant time-delay between their occurence; and contingency relationships, where proximity in time is the main property. We propose an anticipation architecture based on these tools that allows the construction of a relevant internal model of the environment through experience. This internal model can then be used for anticipation and action-decision according to reward expectation. The proposed architecture takes into account the problem of learning complex causal relationships involving many events. We illustrate the effectiveness of the tools proposed with preliminary results about their ability to identify relevant relationships. We conclude by discussing issues not addressed by our model and future investigations.

1 Introduction

Designing agents that can act smartly in a previously unknown environment is one of the most challenging issues in behavioural robotics. Such an agent must have the ability to construct an internal model describing the dynamics of the environment and the effect of its own actions on this environment. This can be mainly understood as extracting causal relationships between events occuring in the environment, whether these events are under the control of the agent (its actions) or if they are externally generated. This internal model allows the agent to predict forthcoming events, as well as the effects of its own actions on the environment. Such a predictive ability paves the way to anticipation and smart decision making by allowing the agent to decide which action to perform to obtain a given outcome. According to the classification of [1], these agents are said to perform *state anticipation*.

Our main focus in this paper is to define and evaluate tools that allow the construction of such an internal model regardless of any reinforcement. In this sense we are very close to latent learning and the concept of *expectancies* proposed by Tolman [6]. We also describe a global architecture for using these tools in a reward based action-decision system but we only sketch out the main guidelines for the reward and action system. On the other hand we will focus on principles of the architecture that allow handling of complex causal relationships involving more than two variables (i.e. combination of events being the cause of a given effect).

The paper is structured as follows: in section 2 we introduce the main information theoretic concepts used in our model and the two kinds of relationships they allow us to identify. Section 3 describes the anticipation architecture embedding these concepts and mechanisms for handling complex causal relationships. In section 4 we describe preliminary results concerning the predictive efficiency of the proposed tools in a simple simulation experiment. Section 5 summarizes the issues our model addresses and we discuss some of those it does not.

2 Information Theory and Anticipation

According to a given past, anticipation can be seen as being able to predict which events will occur in the future and when they will occur. Without any other knowledge, such predictions can only be completely uncertain. The goal of constructing an internal predictive model is to reduce the uncertainty of these predictions. This construction can only be based on information acquired through experience, and therefore on a partial view of the environment, leading to probabilistic representations. Tools for dealing with such representations have been increasingly used in the context of sensorimotor coordination (for example Bayesian modelling in [5]), to analyze properties of the coupling between an agent and its environment (information theoretic approach in [4]) and also to describe conditioning processes with information theory (see [3]). In our context we decided to use information theory because it is an efficient tool to deal with uncertainty and also because recent extensions to this framework seem promising.

2.1 Basis of Information Theory

Shannon's information theory is a mathematical framework that provides quantities about probability distributions of events. We refer the reader to [2] for a complete introduction to the field. One of the main quantities we will be using is the entropy of a probability distribution. Consider a random variable X for which each event x can take a value in the set $\mathcal{X}$. The entropy of this random variable is defined as

$$H(X) = -\sum_{x \in \mathcal{X}} p(x) log_2 p(x) \quad (1)$$

This value reflects the uncertainty about the outcome of this random variable. The minimum is 0 for an absolutely predictable outcome (for example one outcome has a probability of 1) and the maximum is $log_2(|\mathcal{X}|)$ if all outcomes are equiprobable.

The information content or self-information of one particular event x according to a given probability distribution is defined as

$$I(x) = -log_2 p(x) \tag{2}$$

The minimum information content is 0 if this outcome has a probability of 1 and goes toward infinity when the probability reaches 0.
Our use of information theory in this model concerns the extraction of relationships between time-located events such as perceptions or actions. For understanding the tools described below, it is only necessary to keep in mind that high entropy H means high uncertainty, and low information content I means a low probability event (or surprising event).

2.2 Time-delay relationships

We will first focus on time-delay relationships between two events. For example, if an event b always occurs 50 timesteps after another event a, then we would like to identify this relationship. Also we would like the method to have some tolerance for variability, i.e. if b sometimes occurs 49 or 51 timesteps after a, we still consider that there exists a time-delay relationship between them.
For identifying these relationships, we will use information quantities. The principle used is based on the concept of *causal entropy* (see [7]) (which can be interpreted as an easy-to-compute approximation of the information flow model [4]). The idea of causal entropy is the following. Let us consider that we want to identify a time-delay relationship between an event a and an event b always occurring after a. We will then use a random variable $D_{a,b}$ that represents the probability distribution of the time delay between a and b. The entropy of this random variable reflects the strenght of the relationship. The lower the entropy, the stronger the relationship. For example if b always occurs 50 timesteps after a, the entropy of $D_{a,b}$ will be 0 (only one event with a probability of 1, see figure 1).

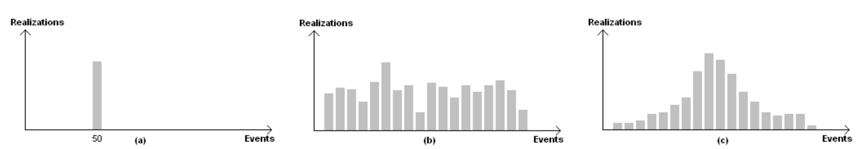

Fig. 1. Histograms of random variables, number of realizations (vertical axis) for each possible event (horizontal axis). **(a)** Histogram of an event b always occuring 50 timesteps after a, $H(D_{a,b}) = -log_2(1) = 0$. **(b)** Example of a high entropy histogram. **(c)** Example of a low entropy histogram.

The original purpose of causal entropy is to determine whether the causal relationship between two events is from a to b or from b to a. This can be determined by comparing the entropies of $D_{a,b}$ and $D_{b,a}$. In our context, the goal is to identify relationships between many events. Therefore, we need a criterion for saying that there exists a time-delay relationship. In [3], the author states that the baseline from which the information provided by a conditional stimulus can be estimated is the prior estimate of the unconditional stimulus frequency. In our framework this can be translated as saying that the criterion for identifying a relationship from a to b is based on the self-relationship $D_{b,b}$, i.e. the average time delay between two successive b events. We will therefore consider that there exists a relationship from a to b if a is a less uncertain predictor for b than b itself, i.e. if

$$H(D_{a,b}) < H(D_{b,b}) \tag{3}$$

Using causal entropy in our context leads to some problems that we need to solve. The first problem is that it is not robust at all to variability in time. If we consider for example two different conditions, in the first one, b occurred 2,10,50 and 100 timesteps after a. In the second case, b occurred 48, 49 ,50 and 51 timesteps after a. For both conditions, $H(D_{a,b}) = 2$ (4 equiprobable outcomes, so $H(D_{a,b}) = log_2(4) = 2$), therefore, we cannot identify which condition reflects a relationship. Obviously the second one seems to be a relationship where b occurs approximatively 50 timesteps after a, whereas the first condition doesn't seem to be a time-delay relationship.

To solve this problem, the idea is to introduce some variability in the probability distribution. Therefore rather than updating the statistics of $D_{a,b}$ by adding one realization of a given time delay t, we add a gaussian distribution of time-delays centered around t, i.e. we add many realizations of t, then a bit less realizations of $t-1$ and $t+1$, even less for $t-2$ and $t+2$, and so on... Now if we get back to our example, adding gaussian realizations of 48, 49, 50 and 51 will lead to overlapping gaussians, and therefore to less variability than in the first condition, and consequently to a lower entropy (see figure 2). For a given time-delay t, the number of realizations to add is computed for growing distances Δ_t as

$$floor\left(\frac{\beta}{\sigma\sqrt{2\pi}} exp\left(-\frac{(\Delta_t - t)^2}{2\sigma^2}\right)\right) \tag{4}$$

until this number reaches 0. The parameters β and σ of this function will be detailed in the Architecture section.

According to the quantity of information gained from using $D_{a,b}$ rather than $D_{b,b}$, we can compute a confidence value of the time-delay expectation as

$$\tau_{a,b} = \frac{H(D_{b,b}) - H(D_{a,b})}{H(D_{b,b})} \tag{5}$$

We can also compute the average expected time-delay between a and b as

$$\delta_{a,b} = \sum_{t \in D_{a,b}} p(t)t \tag{6}$$

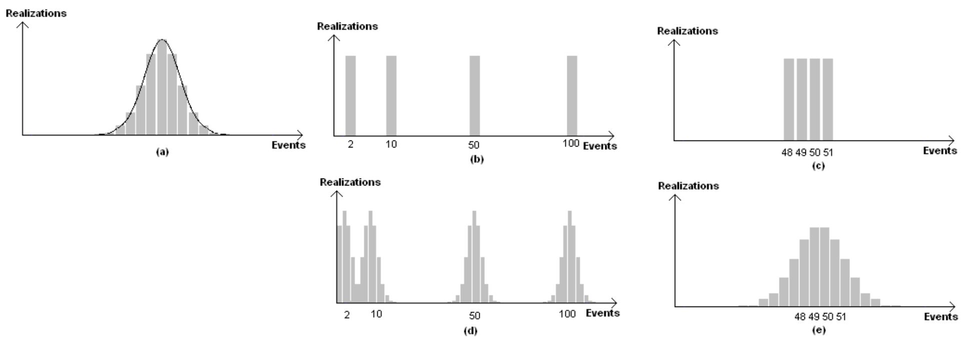

Fig. 2. Usefulness of adding gaussian realizations for time-delay events. **(a)** For one occurrence of a time-delay event, we add a discretized gaussian distribution of realizations centered around the occurring event. **(b)** Example: histogram of the first condition without gaussian realization, $H(D_{a,b}) = 2$. **(c)** Example: histogram of the second condition without gaussian realization, $H(D_{a,b}) = 2$. **(d)** Example: histogram of the first condition with gaussian realization, $H(D_{a,b})$ is high. **(e)** Example: histogram of the second condition with gaussian realization, $H(D_{a,b})$ is low.

Another problem that has to be solved is the following. Let us suppose that after some time we have identified the time-delay relationship between a and b that has been described in the example above (figure 2.e). Now if we consider that a new event c happened 10 timesteps before b, then the histogram of the random variable $D_{c,b}$ would be a perfect gaussian centered on 10. The entropy of this random variable will be lower than the entropy of $D_{a,b}$ because of the small time variation between a and b. But obviously, if we had 4 realizations of b after a (48, 49, 50 and 51 timesteps), then we should be more confident into this relationship than for b after c which had only 1 realization. Put another way, we should be more confident in a relationship that has occurred several times, even with some variation, than into a relationship that occurred only a few times, even with a perfectly constant time-delay. A way to solve this problem is to initialize any random variable $D_{a,b}$ with a uniform probability distribution of time-delays, e.g. an initial white noise. Then multiple realizations of a time-delay, even with some variability, will increase the probability of this time-delay and its neighbourhood, and decrease the probability of the noise values, therefore the entropy of such a random variable will be lower than the entropy of a noisy random variable with only one realization of a time-delay.

2.3 Causal relationships extracted from contingency

Now we will focus on another type of relationship for which there is no precise delay between events a and b. We consider here relationships of the type "when a occurs, b is likely to occur soon". These relationships can be extracted from the contingency of events in the flux of perceptions. We will speak about them as *contingency relationships*, and we will consider that the closer b occurs after a,

the stronger the relationship. Also we will consider that a predicts b if a *mainly* predicts b (relatively to predicting other events) and if b is *mainly* predicted by a (relatively to other events it is predicted by). The purpose of this criterion is the following: let consider an event a than happens all the time, and sometimes an event b, c or d happens. On one hand we can say that b, c and d are well predicted by a, because among all the possible predicting events, a is the most frequent. But on the other hand we cannot say that a predicts correctly b, c or d, because it predicts nearly everything (even itself), and therefore it is a useless predictor. That is why for establishing a predictive relationship from a to b, our criterion takes into account the future of a *and* the past of b.
We can translate these by the following principle: for each event e, we have two random variables, one is related to its past, i.e. it reflects the probability distribution of events that happened before e, we will refer to it as CP_e; and one is related to its future, i.e. the probability distribution of events that happened after e, we will refer to it as CF_e. In this context we will say that there is a relationship between a and b, i.e. that b is a consequence of a if

$$I_{CF_a}(b) < H(CF_a) \tag{7}$$

and

$$I_{CP_b}(a) < H(CP_b) \tag{8}$$

This means that the information carried by b is less than the average information carried by an event that has occurred after a, thus b is more likely to occur after a than other events, and also that a carries less information than the average information carried by an event in the past of b, i.e. a is more likely to have occurred before b than other events.

For each of these variables, event realizations are added according to their distance in time, i.e. when close in time, many realizations of the same event are added (for one true occurrence), the number of realizations added decreasing with the distance. The exact number of realizations follows the same gaussian equation 4, in which we replace t by 0, and Δ_t by the actual distance between the two events (negative values are discarded). Again we can define a confidence value of the contingency expectation, based on the loss of uncertainty, as

$$\kappa_{a,b} = \frac{1}{2}\left(\frac{H(CF_a) - I_{CF_a}(b)}{H(CF_a)} + \frac{H(CP_b) - I_{CP_b}(a)}{H(CP_b)}\right) \tag{9}$$

3 Architecture

The different tools described above are put together in an action-decision architecture based on expectations and reward. In this context we will consider that some perceptions are inherently associated with a reward (negative or positive). The main components of the architecture are the following. First saliency evaluation filters perceptions, forwarding only the unusual ones (those that carry most of the information) or those that are internally associated with a reward.

These perceptions are first stored in short-term memory. Then this short-term memory is used to identify time-delay and contingency relationships in order to build an internal model of the environment and of the agent's interactions. The internal model and the short-term memory are then used together to build expectations about forthcoming events. These three elements are finally used to choose the best action to perform so as to maximize the expected reward. An outline of the architecture is shown in figure 3.

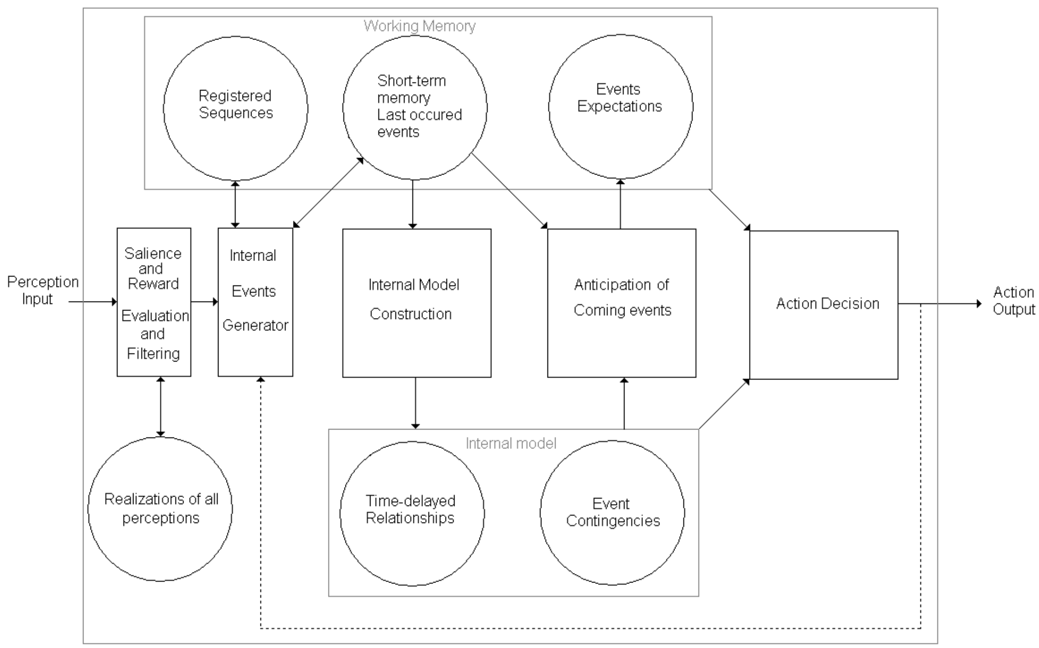

Fig. 3. Main architecture. Circles represent stored information, black boxes are processes that generate information, gray boxes define main blocks. See text for details.

3.1 Construction of the Internal Model

Our main focus in this architecture is its ability to construct the predictive internal model using the information theoretic tools described above. The goal of this predictive model is to account for the dynamics of the environment (so finding relationships between events in the environment) but also to account for the effect of the agent's actions on the environment. The idea to solve this issue in a natural way is to consider any action performed by an agent as en event which is then processed as if it has been generated by the environment.

When an event (or an action) is perceived, it is first stored into short-term memory and replaces any previously stored occurrence of this event. The construction of the internal model is based on the two processes of finding time-delay and contingency relationships. When an event b is processed, for all events a that are in short-term memory, if it is the first occurrence of b since a occurred, we update the statistics of the random variables $D_{a,b}$, CF_a and CP_b. The parameters of the gaussian used for updating the statistics are fixed for $D_{a,b}$ to β_0 and σ_0. For the two other random variables, these are adapted according to the event they concern, i.e. the longer the expected self time-delay between the concerned event, the more the gaussian is flattened. The idea is to adapt to events that occur at very different timescales. Also the β parameter (the heigth of the gaussian) is adapted according to the frequency of the added event, here the idea is to strenghten the association with rare events and to weaken associations with very common events. Therefore when adding and event b to the statistics of a, the parameters used are

$$\sigma = \sigma_0(1 + \alpha\delta_{a,a}) \tag{10}$$

and

$$\beta = \beta_0(1 + \alpha\delta_{a,a} + \lambda\delta_{b,b}) \tag{11}$$

where α is the range adaptation coefficient and λ is the intensity adaptation coefficient (both low positive value). The higher α, the more the gaussian is flattened for a given self time-delay. The higher λ, the more the added event is important for a given self time-delay.

The constructed internal model, along with short-term memory of the last occurred events, can easily be used to determine the expected events using the following principles. For each past event a in the short-term memory, all the $D_{a,b}$ random variables are evaluated, and for each of them which validate the condition 3, the event b is added into the expectation list, along with its average time-delay $\delta_{a,b}$ and its confidence value $\tau_{a,b}$. Then for each possible event b, if we can find any event a in short-term memory that is valid according to contingency conditions 7 and 8, then b is added to the expectations list, again with its average time-delay $\delta_{a,b}$ and its confidence value $\kappa_{a,b}$. Also the outcome of potential actions can be determined by the agent using this internal model.

3.2 Handling complex causal phenomena

One of the most difficult issues of anticipation systems is to be able to identify complex causal phenomena involving many different events. An example of such a phenomena is that when an event a occurs, doing the action b will result in the event c occuring. This issue is tackled in our architecture by the concept of sequence of events. The idea is to construct sequences of events that will be processed as normal events and that can therefore be used as predictors for other events. This process is included in the internal events generator. The problem here is to take care of the combinatorial explosion when grouping events. Therefore we need a criterion for creating new sequences, and also another one for discarding them when they have proved unsuccessful. The idea is to introduce a sequence generation probability p_{sg} that will be used each time an event b

is processed to decide if a new sequence has to be created, another event a is then chosen randomly and a new sequence a, b is registered. Then each time an event is processed, the internal events generator will check if the new event associated with the older ones matches one of the sequences defined. If it is the case, a sequence event is generated and stored in the short-term memory, with the averaged time-distance and saliency of the constituting events. Using a sequence destruction probability p_{sd} evaluated at each time-step, a randomly chosen sequence may be destroyed if it has no predicting power, with a probability growing with the 'age' of this sequence. Forwarding sequence events in the normal events' pathway allows for the construction of longer sequences by associating already existing sequences with other events.

Another case of complex relationship is when an event c predicted by a can be avoided if the action b is performed before c occurs. In this case we have to take into account the NON-occurrence of an expected event. The idea is that when an expected event did not happen after a sufficiently long time, a signal is transmitted to internal event generator which will then generate a special event, opposite of the expected one, and forward it into the normal pathway. For example if an event a predicts an event c, and if after some time this event c still has not arrived, then we will generate an event $\overline{c}$ and forward it into the event processing pathway. This event can then be associated with another event b that caused this non-occurrence, or to the sequence of events a, b.

3.3 Extra Mechanisms

Two other mechanisms are introduced to improve efficiency of the architecture. The first principle is to filter out some of the perceptions to avoid overloading the system with useless information. The precise criterion we use is that according to a distribution probability of perceptions E, which is constantly updated with new perceptions, we consider salient perceptions those that carry more information than the average information carried. Therefore the saliency criterion can be expressed as

$$I(e) > H(E) \tag{12}$$

We also introduce a forgetting mechanism to allow for a quick replacement of relationships that are not relevant anymore. The principle of the forgetting mechanism is to define an upper bound to the total number of realizations of the random variables describing the internal model. When a new realization is added and increases the total number above the defined bound, one other realization is removed, by randomly choosing one of the events stored and removing one realization of this event.

4 Experiments

In this section we will evaluate the ability of the architecture described above to extract relevant causal relationships from the flux of perceptions. The agent is not allowed to act, it can only passively perceive events coming from its environment. We first detail the experimental setup then we analyze the confidence value of relationships of interest.

4.1 Experimental setup

Here we simulate some kind of Skinner box where the agent is situated. The perceptions of the agent are taken from the set $N1, N2, N3, N4, N5, N6, L1, Food$. The events from $N1$ to $N6$ are noise events that have no causal value, whereas events $L1$ and $Food$ are causally associated, $L1$ predicting the $Food$ event ($L1$ stands for *Light 1*, we consider than when the light is flashed, food will be given to the agent in a given delay). $L1 - Food$ sequence has a probability of 0.02 of being initiated at each timestep. The noise events are generated at each timestep with the respective probabilities ($N1 : 0.2, N2 : 0.1, N3 : 0.05, N4 : 0.025, N5 : 0.0125, N6 : 0.00625$). Other parameters of the simulation are the following. Gaussian parameters $\sigma_0 = 3$ and $\beta_0 = 100$. Range adaptation coefficient $\alpha = 0.25$. Intensity adaptation coefficient $\lambda = 0.1$. Random variables have an upper bound of 1000 realizations.
The first experiment measures the confidence values of the contingency and time-delay relationships after 10000 steps of simulation for different time-delay of the $L1 - Food$ association. The time-delays evaluated range from 1 to 80 timesteps with a variability of $+/-3$ timesteps.
For the second experiment we use the same procedure but the parameter investigated is the variability of the time-delay of the $L1 - Food$ association. The base time-delay used is 14 timesteps and with a variability rangind from $+/-0$ to 20 timesteps.
The third experiment aims at evaluating the dynamics of the internal predictive model over time. The $L1 - Food$ association has a time-delay of 14 timesteps and a variability of $+/-3$ timesteps. The experiment is running over 100000 timesteps, and during the range 40000 to 60000 $L1$ and $Food$ are not associated anymore, they are both presented at each timestep with the same probability of 0.01.

4.2 Results

Results of the first and second experiment are shown in figure 4. We can see from these results that contingency relationships are successfully extracted for short time delays, less efficiently when the time delay increases, but they are robust to variability of this time delay. On the other hand, time-delay relationships have the opposite behavior, i.e. they are robust for long time delays, but they loose efficiency as the variability increases. These results confirm the expected behavior of these two anticipation mechanisms, which used together should allow the extraction of most relevant relationships.

Results of the third experiment are shown in figure 5. We can see that both relationships are quickly learned, correctly forgotten when the two stimuli are

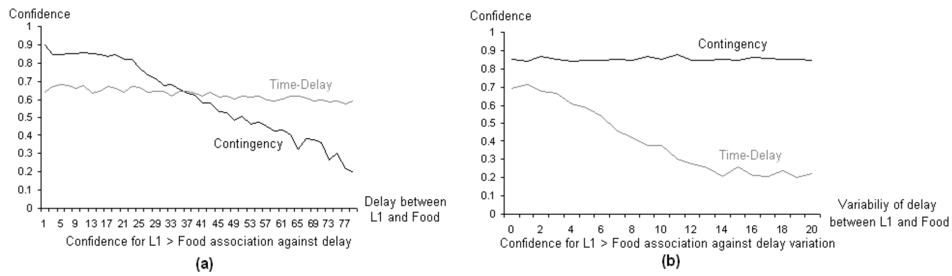

Fig. 4. Plotting of $\kappa_{L1,Food}$ (black) and $\tau_{L1,Food}$ (gray) after 10000 steps simulations. **(a)** Plotting against time delay between $L1$ and $Food$. Time-delay relationship is robust whereas contingency is not. **(b)** Plotting against variability of the time delay between $L1$ and $Food$. Contingency relationship is robust whereas time-delay relationship is not.

not associated anymore, and then their confidence value increases as soon as the events are paired again. These results show that the architecture correctly account for forgetting mechanism. We can see that for a long enough time of exposure to the unpaired events during a typical run, the agent can completely forgot the contingency relationship. On the other hand, the time-delay relationship is maintained for a longer time and its original confidence value is recovered very quickly when the events are paired again, whereas the contingency relationship shows a slower recovery rate.

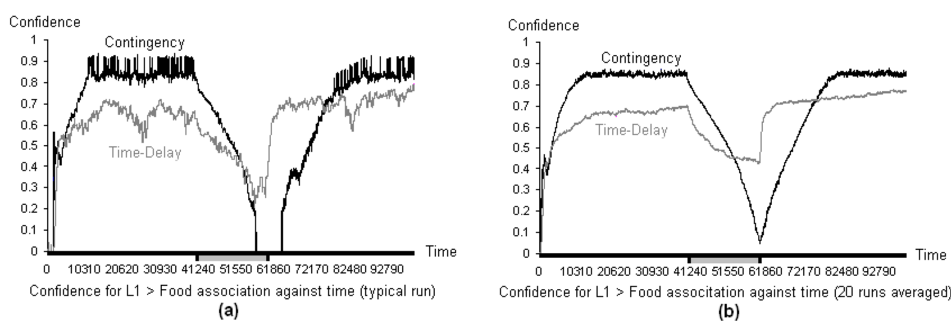

Fig. 5. Plotting of $\kappa_{L1,Food}$ (black) and $\tau_{L1,Food}$ (gray) against time during 100000 steps of simulation. In the range 40000 to 60000 $L1$ and $Food$ are not causally associated (shown in gray on the horizontal axis). **(a)** A typical run. **(b)** Average of 20 experiments.

5 Conclusion

We have introduced two information theory based tools for extracting time-delay and contingency relationships in the flux of perceptions. These tools have been put together into an architecture that uses them for constructing an internal model of the environment. We have shown two distinct properties of contingency and time-delay relationships, the former is robust to variations of the delay between two stimuli, and the latter keeps its efficiency when the time-delay becomes long. We have also shown the efficiency of this architecture for constructing a relevant internal model that is able to quickly adapt to a changing environment.

Some aspects of the architecture could not be detailed here, the first one is its ability to construct and manipulate sequences of events, the second one is the handling of non-occurring expected events, nevertheless the main principles have already been validated. Also as our focus was the construction of the internal model, the action-decision mechanism and reward evaluation could not be investigated, but they have been taken into account in the overall architecture. Experiments illustrating the behaviour generated by the complete architecture will be conducted in future work, and we will especially focus on complex situations involving two-steps learning. Also the architecture does not solve in any way the problem of generalizing learned associations, this could be investigated by introducing concepts already used in learning classifier systems at the level of perceptions.

References

1. Butz, M. V., Sigaud, O., and Gerard,P., (2003). Internal models and anticipations in adaptive learning systems. In Butz, M. V., Sigaud, O., and Gerard,P. (Eds.), Lecture Notes in Computer Science vol. 2684: Anticipatory Behavior in Adaptive Learning Systems. Springer-Verlag.
2. Cover, T., and Thomas, J., (1991). Elements of Information Theory. New York: John Wiley and Sons.
3. Gallistel, C. R., (2002). Frequency, Contingency and the Information Processing Theory of Conditioning. In Sedlmeier, P. and Betsch, T. (Eds.). etc. frequency processing and cognition, pp. 153-171. Oxford University Press.
4. Klyubin, A. S., Polani, D., and Nehaniv, C. L., (2004). Tracking Information Flow through the Environment: Simple Cases of Stigmergy. In Artificial Life IX: Proceedings of the 9th International Conference on the Simulation and Synthesis of Living Systems, pp. 563-568. The MIT Press.
5. Kording, K.P. and Wolpert D.M., (2004). Bayesian integration in sensorimotor learning. Nature 427, pp. 244-247.
6. Tolman, E. C., (1959). Principles of purposive behavior. In Koch, S. (Ed.), Psychology: A study of science pp. 92-157. New York: Mc Graw-Hill.
7. Waddel, J., Dzakpasu, R., Booth, V., Riley, B. T., Reasor, J. D., Poe, G. R., Zochowski, M., (2006). Causal Entropies - a measure for determining changes in the temporal organization of neural systems. Journal of Neuroscience Methods. (submitted)