The Playground Experiment: Task-Independent Development of a Curious Robot

Pierre-Yves Oudeyer and **Frédéric Kaplan** and **Verena V. Hafner** and **Andrew Whyte**
{oudeyer, kaplan, hafner,whyte}@csl.sony.fr
Sony Computer Science Lab, Paris

Abstract

This paper presents the mechanism of Intelligent Adaptive Curiosity. This is an intrinsic motivation system which pushes the robot towards situations in which it maximizes its learning progress. It makes the robot focus on situations which are neither too predictable nor too unpredictable. This mechanism is a source of autonomous mental development for the robot: the complexity of its activities autonomously increases and a developmental sequence appears without being manually constructed. We test this motivation system on a real robot which evolves on a baby play mat with objects that it can learn to manipulate. We show that it first spends time in situations which are easy to learn, then shifts progressively its attention to situations of increasing difficulty, avoiding situations in which nothing can be learnt.

The challenge of autonomous mental development

All humans develop in an autonomous open-ended manner through life-long learning. So far, no robot has this capacity. Building such a robot is one of the greatest challenges to robotics today, and is the long-term goal of the growing field of developmental robotics (Weng *et al.* 2001; Lungarella *et al.* 2003).

There are two characteristic properties of human infant development that can inspire us. First of all, development involves the progressive increase of the complexity of the activities of children with an associated increase of their capabilities. Moreover, infants' activities have always a complexity which is well fitted to their current capabilities. Children undergo a developmental sequence during which each new skill is acquired only when associated cognitive and morphological structures are ready. For example, children learn first to roll over, then to crawl and sit, and only when these skills are operational, they begin to learn how to stand. Development is progressive and incremental. Inspired by this, some roboticists have realized that learning a given task could be made much easier for a robot if it followed a developmental sequence (e.g. "Learning form easy mission" (Asada *et al.* 1996)). But often, the developmental sequence is crafted by hand: roboticists manually build simpler versions of a complex task and put the robot successively in versions of the task of increasing complexity. This technique is useful in many cases, but has shortcomings which limit severely our capacity to build robots that develop in an open-ended manner. Indeed, this is not practical: for each task that one wants the robot to learn, we have to design versions of this task of increasing complexity, and we also have to design manually a reward-function dedicated to this particular task. This might be all right if one is interested in only one or two tasks, but a robot capable of life-long learning should eventually be able to perform thousands of tasks.

This leads us to a second property of child development by which we can be inspired: it is autonomous and active. Of course, adults help by scaffolding their environment, but this is just a help: eventually, infants decide by themselves what they do, what they are interested in, and what their learning situations are. They are not forced to learn the tasks suggested by adults, they can invent their own. Thus, they construct by themselves their developmental sequence. Anyone who has ever played with an infant in its first year knows that for example it is extremely difficult to get the child to play with a toy that is chosen by the adult if other toys and objects are around. In fact, most often the toys that we think are adapted to them and will please them are not at all the one they prefer: they can have much more fun and instructive play experiences with adult objects, such as magazines, keys, or flowers. Also, most of the time infants engage in particular activities for their own sake, rather than as steps towards solving practical problems. This is indeed the essence of play. This suggests the existence of forms of intrinsic motivation, as proposed by psychologists (White 1959) which provide internal rewards during these play experiences. Such internal rewards are obviously useful, since they are incentives to learn many skills that will potentially be readily available later on for challenges and tasks which are not yet foreseeable.

In order to develop in an open-ended manner, robots should certainly be equipped with capacities for autonomous and active development, and in particular with intrinsic motivation systems, forming the core of an architecture for task-independent learning. This crucial topic is still largely an underinvestigated issue. Some researchers have come up with

the idea that this could be achieved by providing the robot with the capability to evaluate operationally concepts such as "novelty", "surprise", "complexity" or degrees of "challenge". The word "curiosity" is often used to denote a system which is able to implement those concepts and search situations of high "novelty" or "challenge". Only a few researchers have suggested to implement mechanisms of artificial curiosity. We will quickly make an overview of the existing systems in the next section. Then, we will present an intrinsic motivation system called Intelligent Adaptive Curiosity, which we developed in order to push some of the limits of existing systems, and to lead a robot towards successive stages of behavioural organization without human intervention.

Existing intrinsic motivation systems

As stated in the last section, existing approaches to intrinsic motivation are all based on an architecture which comprises a machine which learns to anticipate the consequence of the robot's actions, and in which these actions are actively chosen according to some internal measures related to the novelty or predictability of the anticipated situation. Thus, the robots in these approaches can be described as having two modules: 1) one module implements a learning machine **M** which learns to predict the sensorimotor consequences when a given action is executed in a given sensorimotor context; 2) another module is a meta learning machine **metaM** which learns to predict the error that machine **M** makes in its prediction. The existing approaches can be divided into two groups, according to the way action-selection is made depending on the predictions of **M** and **metaM**.

In the first group (Huang & Weng 2002; Thrun 1995; Marshall, Blank, & Meeden 2004) robots directly use the error predicted by metaM to choose which action to do[1]. The action that they choose at each step is the one for which metaM predicts the largest error in prediction of **M**. This has shown to be extremely efficient when the machine **M** has to learn a mapping which is learnable, deterministic and with homogeneous Gaussian noise (Cohn, Atlas, & Ladner 1994; Thrun 1995; Weng 2002; Barto, Singh, & Chentanez 2004). But this method shows limitations when used in a real uncontrolled environment. Indeed, in such a case, the mapping that **M** has to learn is not anymore deterministic, and the noise is vastly inhomogeneous. Practically, this means that a robot using this method will for example be stuck by white noise or situations which are inherently too complex for its learning machinery. For example, a robot equipped with a drive which pushes it towards situations which are maximally unpredictable might discover and stay focused on movement sequences like running fast against a wall, the shock resulting in an unpredictable bounce (in principle, the bounce is predictable since it obeys the deterministic laws of classic mechanics but in practice this prediction requires the perfect knowledge of all the physical properties of the robot body as well as those of the wall, which is typically far from being the case for a robot).

A second group of models tried to avoid getting stuck in the presence of pure noise or unlearnable situations by using more indirectly the prediction of the error of **M** (Schmidhuber 1991; Herrmann, Pawelzik, & Geisel 2000; Kaplan & Oudeyer 2003). Here the action-selection is made not based on the expected error related to the anticipation of the consequences of an action, but on the decrease of this error as compared to a previous prediction of the consequences of action. In brief, an interesting situation is not defined here as a situation of high unpredictability, but as a situation in which the error rate in prediction decreases a lot.

This can be implemented following two possible ideas. In (Herrmann, Pawelzik, & Geisel 2000) and (Kaplan & Oudeyer 2003), the decrease is computed by comparing the expected error in the prediction related to the current action and the mean error related to the predictions about actions which were performed just before the current action. This method has shown interesting results in terms of organization of behaviour (Kaplan & Oudeyer 2003), but is limited. Indeed, in sensorimotor environments in which very different kinds of activities can be performed, such as for example trying to grip an object and trying to vocalize to others, the robot will compare its performances for activities which are of a different kind, which has no obvious meaning. And using a direct measure of the decrease in the error rate in prediction will provide the robot with internal rewards when shifting from an activity with a high mean error rate to activities with a lower mean error rate, which can be higher than the rewards corresponding to an effective increase of the skills of the robot in one of the activities. This will push the robot towards an instable behaviour, in which it focuses on the sudden shifts between different kinds of activities rather than on the actual concentration on activities.

This is why one has to add a mechanism which makes that the robot compares the evolution of its error rates in similar situations, and not necessarily in situations which are contiguous in time. This brings up the generally hard problem of evaluating the similarity of situations. A mechanism has been proposed in (Schmidhuber 1991), but was only tested from an active learning point of view, i.e. how much it allowed to speed up a learning task. Moreover, it was tested in a discrete environment where the similarity of two situations was evaluated by a binary function stating whether they correspond exactly to the same discrete state or not. Finally, this mechanism was compatible only with slow learning algorithms like neural-networks, and not with one-shot learning memory-based algorithms, which are often useful in robotics. In this paper, we will present a system which follows the same basic intuition of evaluating learning progress by comparing the evolution of the error rate in similar situations. Nevertheless, our implementation is quite different, and we will study it from a developmental robotics point of view, i.e. show that it leads to a progressive organization of the behaviour of the robot. In particular, we will show how our system leads to the autonomous formation of a developmental sequence comprising more than one stage. To our knowledge, other systems based on artificial curiosity were not already shown to do that: indeed, typically they have

[1] Of course, we are only talking about the "novelty" drive here: their robots are sometimes equipped with other competing drives or can respond to external human based reward sources

allowed for the development and emergence of one level of behavioural patterns, but did not show how new levels of more complex behavioural patterns could emerge without an intervention of a human or a change in the environment provoked by a human. Moreover, we will test the mechanism in a real robotic set-up with a 5-dimensional continuous motor space in which evaluating the similarity of situations is non trivial.

Intelligent Adaptive Curiosity

The intrinsic motivation system that we developed is called **Intelligent Adaptive Curiosity** (Oudeyer & Kaplan 2004), which can be abbreviated as IAC.

Summary. IAC relies on a memory which stores all the experiences encountered by the robot in the form of vector exemplars. There is a mechanism which incrementally splits the sensorimotor space into regions, based on these exemplars. Each region is characterized by its exclusive set of exemplars. Each region is also associated with its own learning machine, which we call an expert. This expert is trained with the exemplars available in its region. When a prediction corresponding to a given situation has to be made by the robot, then the expert of the region which covers this situation is picked up and used for the prediction. Each time an expert makes a prediction associated to an action which is actually executed, its error in prediction is measured and stored in a list which is associated to its region. Each region has its own list. This list is used to evaluate the potential learning progress that can be gained by going in a situation covered by its associated region. This is made based on a smoothing of the list of errors, and on an extrapolation of the derivative. When in a given situation, the robot creates a list of possible actions and chooses the one for which it evaluates it will lead to a situation with maximal expected learning progress. Technical details are provided in the appendix.

The Playground Experiment: the discovery of sensorimotor affordances

In a previous paper (Oudeyer & Kaplan 2004), we presented an implementation of this system in a simulated robot. We showed how IAC could allow the robot to develop in a noisy inhomogeneous environment, without being trapped by noise or the alternation between very unpredictable and very predictable situations. However, this experiment was in a simulated environment, and its complexity was limited.

The experimental set-up presented in this paper is called "The Playground Experiment". This involves a physical robot as well as a more complex sensorimotor system and environment. We use a Sony AIBO robot which is put on a baby play mat with various toys that can be bitten, bashed or simply visually detected (see fi gure 1). We have developed a web site which presents pictures and videos of this set-up: http://playground.csl.sony.fr/.

Motor control. The robot is equipped with three basic motor primitives: turning the head, bashing and crouch biting. Each of them is controlled by a number of real number parameters, which are the action parameters that the robot controls. The "turning head" primitive is controlled with the pan and tilt parameters (p and t) of the robot's head. The "bashing" primitive is controlled with the strength and the angle (b_s and b_a) of the leg movement (a lower-level automatic mechanism takes care of setting the individual motors controlling the leg). The "crouch biting" primitive is controlled by the depth of crouching d (and the robot crouches in the direction in which it is looking at, which is determined by the pan and tilt parameters). To summarize, choosing an action consists in setting the parameters of the 5-dimensional continuous vector $\mathbf{M}(\mathbf{t})$:

Figure 1: The Playground Experiment

$$\mathbf{M}(\mathbf{t}) = (p,\ t,\ b_s,\ b_a,\ d)$$

All values are real numbers between 0 and 1, plus the value -1 which is a convention used for not using a motor primitive.

Perception. The robot is equipped with three high-level sensors based on lower-level sensors. The sensory vector $\mathbf{S}(\mathbf{t})$ is thus 3-dimensional:

$$\mathbf{S}(\mathbf{t}) = (O_v,\ B_i,\ O_s)$$

where O_v is the binary value of an object visual detection sensor using the video camera of the AIBO, B_i is the binary value of a biting sensor, based on the cheek sensor that the AIBO possess, and O_s is the binary value of an oscillation sensor based on the infra-red distance sensor of the AIBO.

Initially the robot knows nothing about sensorimotor affordances. For example, it does not know that the values of the object visual detection sensor are correlated with the values of its pan and tilt. It does not know that the values of the biting or object oscillation sensors can become 1 only when biting or bashing actions are performed towards an object. It does not know that some objects are more prone to provoke changes in the values of the B_i and O_s sensors when only certain kinds of actions are performed in their direction. It does not know for example that to get a change in the value of the oscillation sensor, bashing in the correct direction is not enough, because it also needs to look in the right direction (since its oscillation sensors are on the front of its head). These remarks allow to understand easily that a random strategy will not be effi cient in this environment. If the robot would do random action selection, in a vast majority of cases nothing would happen (especially for the B_i and O_s sensors).

The action perception loop. To summarize, the mapping that the robot learns is:

$$f:\ \mathbf{SM(t)} = (p,\ t,\ b_s,\ b_a,\ d,\ O_v,\ B_i,\ O_s)$$
$$\longmapsto \mathbf{S(t+1)} = (\widetilde{O_v},\ \widetilde{B_i},\ \widetilde{O_s})$$

The robot is equipped with the Intelligent Adaptive Curiosity system, and thus chooses its actions according to the potential learning progress that it can provide to one of its experts.

Results

During an experiment we continuously measure a number of features which help us to characterize the dynamics of the robot's development. First, we measure the frequency of the different kinds of actions that the robot does in a given time window. More precisely, every 100 actions we measure: 1) the percentage of actions in the last 100 actions which do not involve the biting and the bashing motor primitive (i.e. the robot's action boils down to "just looking" in a given direction); 2) the percentage of actions in the last 100 actions which involve the biting motor primitive; 3) the percentage of actions in the last 100 actions which involve the bashing motor primitive. Second, we measure the distribution of values in each of the three sensory channels O_v, B_i and O_s, every 100 actions and during the last 100 actions and we normalize these values by the distribution of the corresponding values in the case of random action selection. We normalize with the corresponding values of the random action selection method in order to show more clearly that some interesting and complex behaviours which are extremely rare with random action selection may become quite frequent when using Intelligent Adaptive Curiosity.

We will now show details of an example for a typical run of the experiment. All the curves corresponding to the measures we described are in figure 2. From the careful study of these curves, augmented with the study of the trace of all the situations that the robot encountered, we observe that 1) there is an evolution in the behaviour of the robot; 2) this evolution is characterized by qualitative changes in this behaviour; 3) these changes correspond to a sequence of more than two phases of increasing behavioural complexity, i.e. we observe the emergence of several successive levels of behavioural patterns. It is possible to summarize the evolution of these behavioural patterns using the concept of stages, where a stage is here defined as a period of time during which some particular behavioural patterns occur significantly more often than random and did not occur significantly more often than random in previous stages. These behavioural patterns correspond to combinations of clear deviations from the mean in the curves in figure 2. Here are the different stages which are visually denoted in figure 2 with letters $P_1, ..., P_5$:

Stage 1: the robot has a short initial phase of random exploration and body babbling. This is because during this period there are few experts yet and so the sensorimotor space has not yet been partitioned in significantly different areas;

Stage 2: the robot stops using the biting and bashing primitives, and spends most of its time looking around. It has discovered that at this stage of its development, this kind of action is the greatest source of learning progress. The study of the curve measuring the distribution of O_v values shows that it does not see objects very often: it is in fact spending time learning that in many areas of the space there are no objects;

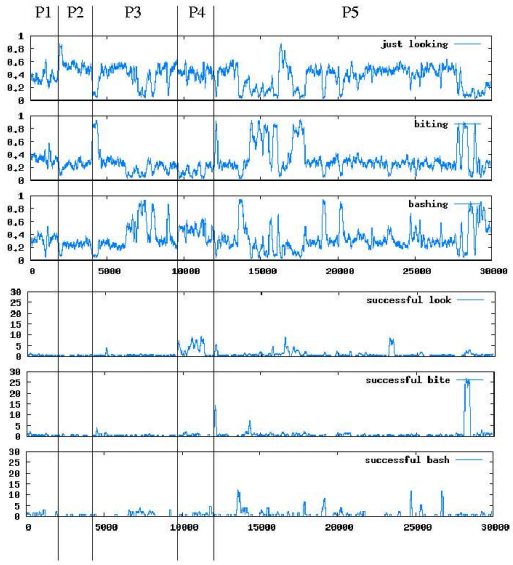

Figure 2: Top 3: Frequency for certain action types on windows 100 time steps wide. Top: bashing. Centre: Biting. Bottom: Just looking. Bottom 3: Distribution of values of the three sensors (O_v, B_i, O_s) on windows 100 time steps wide, normalised regarding to the distribution of values in the case of random action selection.

Stage 3: then there is a phase during which the robot begins to use a lot the biting and bashing primitives. It discovers that using these primitives sometimes produces something. Yet, the curve measuring the distribution of O_v values as well as the close inspection of the traces of the experiment shows again that the robot is not oriented often towards objects: this means that it has not discovered yet the fact that there is a relation both among the motor primitive (e.g. looking in the same direction as the movement of the bashing) and among action primitives and external objects (e.g. that biting or bashing can produce a result only if applied to an object);

Stage 4: then the robot discovers a new niche of learning progress at this stage of its development: it now starts to look often towards objects, as shown by the O_v curve. Yet, it is now half of the time stopping its action, and the rest of the time often bashing and sometimes biting, but with no specific association between the type of action (biting or bashing) and the objects towards which it is directed (the biteable or the bashable object). This means that the robot is here learning the precise location of objects as well as the fact that doing "something" towards an object can sometimes produce a reaction on the object

and on its sensors.

Stage 5: Finally, the robot comes into a phase in which it discovers the precise affordances between action types and particular objects: it is at this stage focusing either on trying to bite the biteable object, and on trying to bash the bashable object, as we can deduce from the curves showing the distribution of B_i and O_s values. It is striking to note that during this phase, there are periods of time during which these coordinated motor primitives towards the right associated objects are 30 times more frequent as compared to the frequency of these situations in the random action selection case. Furthermore, it does actually manage to bite and bash successfully quite often, which of course is an emergent side effect of Intelligent Adaptive Curiosity and was not a pre-programmed task.

We made several experiments and each time we got a similar structure in which a self-organized developmental sequence pushed the robot towards activities of increasing complexity, in particular towards the progressive discovery of the sensorimotor affordances of various levels of detail. Nevertheless, we also observed that two developmental sequences are never exactly the same, and the number of phases sometimes changes a bit or intermediary phases are sometimes exchanged. It is interesting to note that this is also true for children: for example, some of them learn to crawl before they can sit, and vice versa. We are now trying to make statistical measures about the set of developmental sequences that are generated in our experiments in order to understand better how particular environment and embodiment conditions lead to the formation of recurrent developmental stages.

Acknowledgements

We would like to thank Jean-Christophe Baillie for his URBI system (Baillie 2004) which we used to program the robot and Luc Steels for precious comments on this work. This research has been partially supported by the ECAGENTS project founded by the Future and Emerging Technologies programme (IST-FET) of the European Community under EU R&D contract IST-2003-1940.

Appendix

Sensorimotor apparatus. The robot has a number of real-valued sensors $s_i(t)$ which are here summarized by the vector $\mathbf{S(t)}$. Its actions are controlled by the setting of the real number values of a set of action/motor parameters $m_i(t)$, which we summarize using the vector $\mathbf{M(t)}$. We denote the sensorimotor context $\mathbf{SM(t)}$ as the vector which summarizes the values of all the sensors and the action parameters at time t (it is the concatenation of $\mathbf{S(t)}$ and $\mathbf{M(t)}$). In all that follows, there is an internal clock in the robot which discretized the time, and new actions are chosen at every time step.

Regions. IAC equips the robot with a memory of all the exemplars $(\mathbf{SM(t)}, \mathbf{S(t+1)})$ which have been encountered by the robot. There is a mechanism which incrementally splits the sensorimotor space into regions, based on these exemplars. Each region is characterized by its exclusive set of exemplars. At the beginning, there is only one region $\mathcal{R}_1$. Then, when a criterion C_1 is met, this region is split into two regions. This is done recursively. A simple criterion C_1 can be used: when the number of exemplars associated to the region is above a threshold $T = 250$, then split. This criterion allows computational efficiency.

When a splitting has been decided, then another criterion C_2 must be used to find out how the region will be split. This criterion splits the set of exemplars into two sets so that the sum of the variances of $\mathbf{S(t+1)}$ components of the exemplars of each set, weighted by the number of exemplars of each set, is minimal.

Recursively and for each region, if the criterion C_1 is met, the region is split into two regions with the criterion C_2. Each region stores all the cutting dimensions and the cutting values that were used in its generation as well as in the generation of its parent experts. As a consequence when a prediction has to be made of the consequences of $\mathbf{SM(t)}$, it is easy to find out the expert specialist for this case: it is the one for which $\mathbf{SM(t)}$ satisfies all the cutting tests.

Experts. To each region $\mathcal{R}_n$, there is an associated learning machine $\mathbf{E_n}$, called an expert. A given expert $\mathbf{E_n}$ is responsible for the prediction of $\mathbf{S(t+1)}$ given $\mathbf{SM(t)}$ when $\mathbf{SM(t)}$ is a situation which is covered by its associated region $\mathcal{R}_n$. Each expert $\mathbf{E_n}$ is trained on the set of exemplars which is possessed by its associated region $\mathcal{R}_n$. An expert can be a neural-network, a support-vector machine or a Bayesian machine for example. When a region is split, two child experts are created as fresh experts re-trained with the exemplars that their associated region has inherited.

Evaluation of learning progress. This partition of the sensorimotor space into different regions is the basis of our regional evaluation of learning progress. Each time an action is executed by the robot in a given sensorimotor context $\mathbf{SM(t)}$ covered by the region $\mathcal{R}_n$, the robot can measure the discrepancy between the sensory state $\widetilde{\mathbf{S}}\mathbf{(t+1)}$ that the expert $\mathbf{E_n}$ predicted and the actual sensory state $\mathbf{S(t+1)}$ that it measures. This provides a measure of the error of the prediction of $\mathbf{E_n}$ at time $t+1$:

$$e_n(t+1) = ||\mathbf{S(t+1)} - \widetilde{\mathbf{S}}\mathbf{(t+1)}||^2$$

This squared error is added to the list of past squared errors of $\mathbf{E_n}$, which are stored in association to the region $\mathcal{R}_n$. We denote this list:

$$e_n(t), e_n(t-1), e_n(t-2), ..., e_n(0)$$

Note that here t denotes a time which is specific to the expert, and not to the robot: this means that $e_n(t-1)$ might correspond to the error made by the expert $\mathbf{E_n}$ in an action performed at $t-10$ for the robot, and that no actions corresponding to this expert were performed by the robot since that time. These lists associated to the regions are then used to evaluate the learning progress that has been achieved after an action $\mathbf{M(t)}$ has been achieved in sensory context $\mathbf{S(t)}$, leading to a sensory context $\mathbf{S(t+1)}$. The learning progress that has been achieved through the transition from the $\mathbf{SM(t)}$ context, covered by region $\mathcal{R}_n$, to the context with a perceptual vector $\mathbf{S(t+1)}$ is computed as the smoothed derivative of the error curve of $\mathbf{E_n}$ corresponding to the acquisition of its recent exemplars. Mathematically, the computation involves two steps:

- the mean error rate in prediction is computed at $t+1$ and $t+1-\tau$:

$$< e_n(t+1) >= \frac{\sum_{i=0}^{\theta} e_n(t+1-i)}{\theta+1}$$

$$< e_n(t+1-\tau) >= \frac{\sum_{i=0}^{\theta} e_n(t+1-\tau-i)}{\theta+1}$$

  where τ is a time window parameter typically equal to 15, and θ a smoothing parameter typically equal to 25.

- the actual decrease in the mean error rate in prediction is defined as $D(t+1) =< e_n(t+1) > - < e_n(t+1-\tau) >$. We can then define the actual learning progress as

$$L(t+1) = -D(t+1)$$

Eventually, when a region is split into two regions, both new regions inherit the list of past errors from their parent region, which allows them to make evaluation of learning progress right from the time of their creation.

Action selection. We have now in place a prediction machinery and a mechanism which provides an internal reward (positive or negative) $r(t) = L(t)$ each time an action is performed in a given context, depending on how much learning progress has been achieved. The goal of the intrinsically motivated robot is then to maximize the amount of internal reward that it gets. Mathematically, this can be formulated as the maximization of future expected rewards (i.e. maximization of the return), that is $E\{\sum_{t \geq t_n} \gamma^{t-t_n} r(t))\}$ where γ $(0 \leq \gamma \leq 1)$ is the discount factor, which assigns less weigh on the reward expected in the far future.

This formulation corresponds to a reinforcement learning problem formulation (Sutton & Barto 1998) and thus the complex techniques developed in this field can be used to implement an action selection mechanism which will allow the robot to maximize future expected rewards efficiently. Yet, the purpose of this article is to focus on the study and understanding of the learning progress definition that we presented. Using a complex re-inforcement machinery brings complexity and biases which are specific to a particular method, especially concerning the way they process delayed rewards. While using such a method with intrinsic motivation systems will surely be useful in the future, and is in fact an entire subject of research as illustrated by the work described in (Barto, Singh, & Chentanez 2004), we will make a simplification which will allow us not to use such sophisticated re-inforcement learning methods so that the results we will present in the experiment section can be interpreted more easily. This simplification consists in having the system try to maximize only the expected reward it will receive at $t+1$, i.e. $E\{r(t+1)\}$ This permits to avoid problems related to delayed rewards and it makes it possible to use a simple prediction system which can predict $r(t+1)$, and so evaluate $E\{r(t+1)\}$, and then be used in a straightforward action selection loop. The method we use to evaluate $E\{r(t+1)\}$ given a sensory context $\mathbf{S(t)}$ and a candidate action $\widetilde{\mathbf{M}}(\mathbf{t})$, constituting a candidate sensorimotor context $\widetilde{\mathbf{SM}}(\mathbf{t})$ covered by region $\mathcal{R}_n$, is straightforward but revealed to be efficient: it is equal to the learning progress that was achieved in $\mathcal{R}_n$ with the acquisition of its recent exemplars, i.e. $E\{r(t+1)\} \approx L(t - \theta_{R_n})$ where $t - \theta_{R_n}$ is the time corresponding to the last time region $\mathcal{R}_n$ and expert $\mathbf{E_n}$ processed a new exemplar.

Based on this predictive mechanism, one can deduce a straightforward mechanism which manages action selection in order to maximize the expected reward at $t+1$:

- in a given sensory $\mathbf{S(t)}$ context, the robot makes a list of the possible actions $\widetilde{\mathbf{M}}(\mathbf{t})$ which it can do; If this list is infinite, which is often the case since we work in continuous action spaces, a sample of candidate actions is generated;
- each of these candidate actions $\widetilde{\mathbf{M}}(\mathbf{t})$ associated with the context makes a candidate $\widetilde{\mathbf{SM}}(\mathbf{t})$ vector for which the robot finds out the corresponding region $\mathcal{R}_n$; then the formula we just described is used to evaluate the expected learning progress $E\{r(t+1)\}$ that might be the result of executing the candidate action $\widetilde{\mathbf{M}}(\mathbf{t})$;
- the action for which the system expects the maximal learning progress is chosen and executed except in some cases when a random action is selected. In the following experiments ϵ is typically 0.35.
- after the action has been executed and the consequences measured, the system is updated.

References

Asada, M.; Noda, S.; Tawaratsumida, S.; and Hosoda, K. 1996. Purposive behavior acquisition on a real robot by vision-based reinforcement learning. *Machine Learning* 23:279–303.

Baillie, J. 2004. Urbi: A universal language for robotic control. *International Journal of Humanoid Robotics*.

Barto, A.; Singh, S.; and Chentanez, N. 2004. Intrinsically motivated learning of hierarchical collections of skills. In *3rd International Conference on Development and Learning*.

Cohn, D.; Atlas, L.; and Ladner, R. 1994. Improving generalization with active learning. *Machine Learning* 15(2):201–221.

Herrmann, J.; Pawelzik, K.; and Geisel, T. 2000. Learning predicitve representations. *Neurocomputing* 32-33:785–791.

Huang, X., and Weng, J. 2002. Novelty and reinforcement learning in the value system of developmental robots. In Prince, C.; Demiris, Y.; Marom, Y.; Kozima, H.; and Balkenius, C., eds., *Proceedings of the 2nd international workshop on Epigenetic Robotics : Modeling cognitive development in robotic systems*, 47–55. Lund University Cognitive Studies 94.

Kaplan, F., and Oudeyer, P.-Y. 2003. Motivational principles for visual know-how development. In Prince, C.; Berthouze, L.; Kozima, H.; Bullock, D.; Stojanov, G.; and Balkenius, C., eds., *Proceedings of the 3rd international workshop on Epigenetic Robotics : Modeling cognitive development in robotic systems*, 73–80. Lund University Cognitive Studies 101.

Lungarella, M.; Metta, G.; Pfeifer, R.; and Sandini, G. 2003. Developmental robotics: A survey. *Connection Science* 15(4):151–190.

Marshall, J.; Blank, D.; and Meeden, L. 2004. An emergent framework for self-motivation in developmental robotics. In *3rd International Conference on Development and Learning*.

Oudeyer, P.-Y., and Kaplan, F. 2004. Intelligent adaptive curiosity: a source of self-development. In Berthouze, L.; Kozima, H.; Prince, C. G.; Sandini, G.; Stojanov, G.; Metta, G.; and Balkenius, C., eds., *Proceedings of the 4th International Workshop on Epigenetic Robotics*, volume 117, 127–130. Lund University Cognitive Studies.

Schmidhuber, J. 1991. Curious model-building control systems. In *Proceeding International Joint Conference on Neural Networks*, volume 2, 1458–1463. Singapore: IEEE.

Sutton, R., and Barto, A. 1998. *Reinforcement learning: an introduction*. Cambridge, MA.: MIT Press.

Thrun, S. 1995. Exploration in active learning. In Arbib, M., ed., *Handbook of Brain Science and Neural Networks*. MIT Press.

Weng, J.; McClelland, J.; Pentland, A.; Sporns, O.; Stockman, I.; Sur, M.; and Thelen, E. 2001. Autonomous mental development by robots and animals. *Science* 291:599–600.

Weng, J. 2002. A theory for mentally developing robots. In *Second International Conference on Development and Learning*. IEEE Computer Society Press.

White, R. 1959. Motivation reconsidered: The concept of competence. *Psychological review* 66:297–333.