5th International Conference on Development and Learning
and on Epigenetic Robotics
Aug 13-16, 2015. Providence, RI USA

Which criteria for autonomously shifting between goal-directed and habitual behaviors in robots?

Erwan Renaudo*†, Benoît Girard*†, Raja Chatila*†, Mehdi Khamassi*†
*Sorbonne Universités, UPMC Univ Paris 06,
UMR 7222, Institut des Systèmes Intelligents et de Robotique, F-75005, Paris, France
†CNRS, UMR 7222, Institut des Systèmes Intelligents et de Robotique, F-75005, Paris, France
renaudo@isir.upmc.fr

Abstract—**Research in the fields of Psychology and Neuroscience have provided strong evidence that mammals can adaptively switch between goal-directed behaviors - i.e. deliberative decisions based on costly but flexible planned long-term consequences of actions - and habitual behaviors - i.e. reactive behaviors that are efficient when the environment is stable but inflexible in the case of environmental changes. However, the computational principles underlying this switching ability are not yet understood, and several alternative criteria have been proposed, each tested on specific subsets of experimental datasets. Here we present a neurorobotic implementation and comparison of such type of criteria, plus some new ones imported from the field of ensemble reinforcement learning, with a two-fold objective: on the one hand exploring the possible efficiency of such bio-inspired principles to enable robots to have more behavioral flexibility during autonomous development and learning; on the other hand, analyzing whether an asynchronous continuous robotic simulation and comparison of these criteria in a common task can feed current debates in the Psychological and Neuroscience fields. We evaluate these methods in an apparently simple repetitive cube-pushing task on a simulated conveyor belt, but which imposes to the robot constant trade-offs between speed and accuracy and between stability and abrupt changes. Our results show that if overall performance is not improved by using multiple behavioral systems in a stable environment, these methods allow for a better adaptation to environmental changes. The Voting methods and Boltzmann addition, from ensemble reinforcement learning, give the best performance, providing an interesting alternative to Expert selection.**

I. Introduction

Studies of behavior in mammals have highlighted two main kinds of behaviors during decision-making tasks: goal-directed behaviors governed by estimates of action-outcome contingencies are mainly active at the beginning of the task, while a transfer of control to habitual behaviors governed by stimulus-response associations occurs when the animal is extensively trained in the task under stable conditions [1]. Dolan et al. review these models in [2]. Goal-directed behaviors allow the animal to be sensible to outcome devaluation and to flexibly adapt to new conditions (e.g. avoid food that has been poisoned). Habitual behavior is characterized by the animal persevering in its behavior even after outcome devaluation [3][4]. On the other hand, goal-directed behaviors are hypothetized as slow and costly before making a decision while habits allow the animal to perform quickly and efficiently in a familiar task and environment [5]. These behaviors are modeled using the theory of Reinforcement Learning (RL) [6]: model-based and model-free algorithms (here called "Experts") provide a direct analogy with goal-directed and habitual behaviors [7]. Different computational criteria have been proposed to decide when to shift between model-based and model-free Experts. Applied to neuroscience tasks, the work from Daw et al. [7] proposes that the most certain Expert gets control on the agent, while Keramati [8] balances speed and accuracy using the cost of planning versus the gain of information. A third approach proposes, in the context of navigation strategies, that an arbitration module learns by reinforcement the most efficient behavior (in terms of average obtained reward) in each state [9]. It has been successfully applied to robotics [10], however it suffers from the RL algorithms intrinsic properties, namely long learning and slow adaptation of policy to changes for model-free algorithms and cost of planning for model-based algorithms. Thus the contribution of this article is to compare different existing criteria for the arbitration between MB and MF reinforcement learning in the same robotic experiment extended from our previous work [11] where we only tested MF only, MB only and a random combination of the two.

978-1-4673-9319-5/15/$31.00 ©2015 IEEE

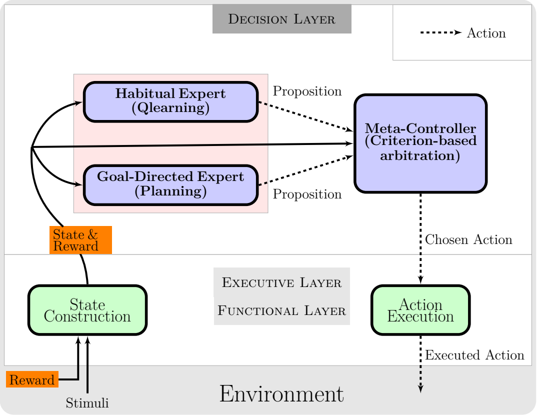


Figure 1: *Global Architecture with habit learning Decision Layer ; From stimuli received, an abstract state is build up, associated with a reward depending on previous state and action and sent to the Experts and the Meta-Controller. From this information, each Expert learns and decides what action to execute in this state. These propositions are sent to the Meta-Controller that arbitrate to decide what will be the final action.*

II. Control Architecture

In this work, we extend the model proposed in [11] where the third layer (i.e. Decisional layer) of a classical robotic three-layered architecture [12] is extended to coordinate a model-based Reinforcement Learning algorithm (Value Iteration) as Goal-Directed Expert (MB) and a model-free Reinforcement learning algorithm (Qlearning) as Habitual Expert (MF) (fig. 1). Each of these Experts takes a decision on which action $a \in \mathcal{A}$ to execute in the current state $s \in \mathcal{S}$, and a Meta-Controller (MC) selects one of the propositions – based on criteria detailed below – to be executed. Experts are implemented as in [11], especially for the MB features. In this version, the Meta-Controller is receiving propositions from each Expert instead of arbitrating *a priori* only with the state information. This organisation allows to get internal Expert information on the way the decision is taken (i.e. the final action probability distribution from which the decision is drawn). Each Expert computes its proposition when the state information is received, but they are received asynchronously by the Meta-Controller, due to the time needed for planning by the MB Expert. However, as the total time allowed to the latter to take a decision is bounded, we choose to wait for each Expert proposal before arbitration. We studied two kinds of arbitration, the first based on signals from the task, the second based on Expert proposition fusion.

A. Tested criteria

1) Signals: From Experts running individually, relevant information on the task and the environment can be extracted. We analyse (a) a particular measure of uncertainty corresponding to the entropy of action probability distribution for each Expert (fig. 2a) and (b) a measure of instantaneous performance corresponding to the average reward received over time (fig. 2b).

The entropy measures how peaked is the distribution, thus giving an estimation of how confident is each Expert in its decisions. Based on this information on uncertainty, we can choose the most certain Expert's proposition. In every state where an Expert has learnt the most rewarding action, the entropy will be lower than in states where it has no clue on the action to do. However, this information can sometimes be misleading, especially when the environment or the goal has changed after extensive learning: the Habitual Expert is long to adapt, and will keep a low entropy even when it's behavior is not adapted anymore.

Besides, we can monitor the mean reward (cf. fig 2b) received by the agent. The agent can get any value of reward - by executing action a in state s - from the environment, adequate behaviors being positively rewarded and leading to a high mean reward. This information tells us how adapted is the behavior :

- if the mean reward is constant, its value provides a relative estimation of how good is policy. It should be the case after convergence of learning in a stable environment.
- if the mean reward is increasing, we can deduce that the agent is currently discovering a better policy, or that environment has changed such that the current policy has become more relevant.
- if the mean reward is decreasing, the agent's behavior is not adapted anymore after an environmental change.

We estimate incrementally the mean reward using exponential decay (α : smoothing factor):

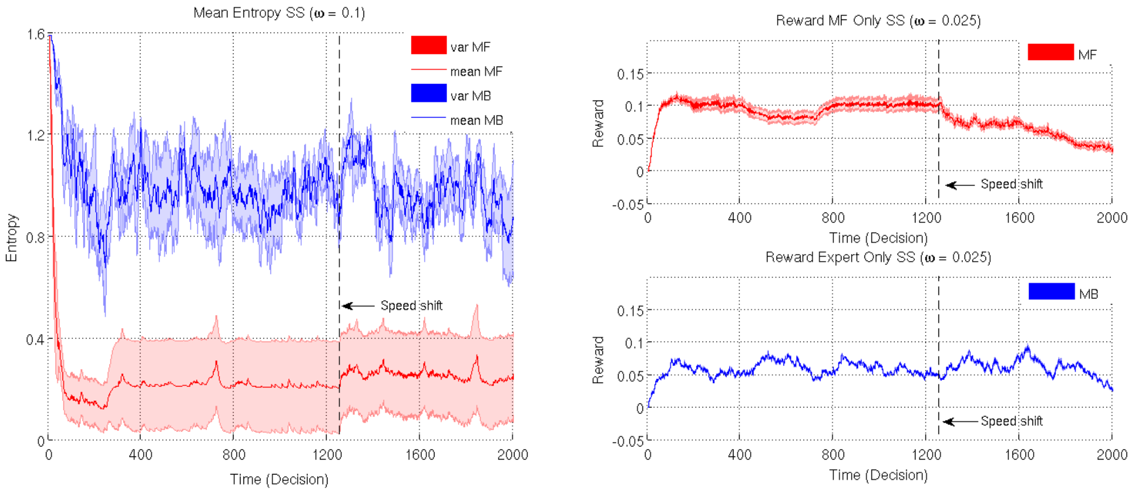
$$\bar{r}_t = (1-\alpha) * \bar{r}_{t-1} + \alpha * r_t \quad (1)$$

and the entropy of Expert E as the Shannon entropy (with $P_i = p(a = a_i|s)$):

$$H_t^E(x) = -\sum_{i=0}^{|\mathcal{A}|} P_i * log_2(P_i) \quad (2)$$

The used criteria are:

- *Entropy*: the Meta-Controller follows the proposition of the most confident Expert, i.e. the Expert with the lowest entropy.



(a) Action distribution entropy.

(b) Mean reward.

Figure 2: *Expert action Entropy and Mean reward evolution in 10 runs of Speed Shift cases of the Block Push experiment (see III-A) when the agent is controlled* ***by each Expert alone****. For analysis, data from these experiments are filtered by a low-pass filter of parameter ω. After learning a policy, mean reward and entropy have reached a stable level. Entropy shows which Expert is the most confident in its proposal, which can be used as an arbitration criterion. When the environment changes, making the current policy less efficient, the mean reward for MF Expert drops (fig 2b), indicating the shift in condition, while the MB Expert is more robust to such changes. A corresponding increase in both entropies can be seen (fig. 2a).*

- *Mean Reward*: the Meta-Controller follows the proposition of the Habitual Expert if the mean reward is increasing, of the Goal-Directed Expert otherwise.

The second criterion intends to give control to the Goal-Directed Expert in cases where performance is dropping, which may signal a change in the environment, and the need to rely on an Expert that can quickly react to this change.

2) Proposition fusion: We also test different strategies for merging action probability distributions from the two Experts, from Ensemble Reinforcement Learning proposed in [13]. Instead of arbitrating between propositions, a probability distribution is computed – by merging Experts knowledge – from which the final decision is taken. Four merging methods are tested, to value each action :

- *Majority vote:* the most probable action of each Expert receives a value of 1, others receive 0. In our implementation, we give a value of $1/n, n$ being the number of equally most probable actions, so that we can give less importance to an expert that is uncertain about the most relevant action. These values are then summed over Experts to shape the final probability distribution.
- *Rank vote:* actions are ranked highest-first depending on their probability and given a decreasing value depending on the rank. These values are summed over Experts to give the final action value.
- *Boltzmann Multiplication:* the final action values are computed as the product of their probabilities over each Expert.
- *Boltzmann Addition:* the final action values are computed as the sum of their probabilities over each Expert.

For the two voting-based criteria, the final values are converted into probabilities using a Softmax function. For Boltzmann operations, the final probabilities of action come from normalizing the distribution. As reference point we also tested a Random criterion proposed in [11] which chooses randomly between MF and MB at each iteration in order to simply assess the usefulness of combining two Experts rather than using just one. The architecture is implemented with ROS [14]. Thus no synchrony is forced between the Experts, and perceptions, states and actions are flowing asynchrously – depending of the duration of processing the information – through the architecture, as it will on a real robot implementation.

III. Results

A. Experiment Description

The architecture is evaluated in the same simulated Block-Push task than described in [11] (see fig. 3).

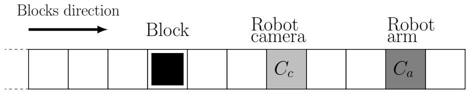


Figure 3: *The experimental setup described in [11]: blocks are carried on a discrete conveyor belt in front of the robot, at a certain speed. C_c is the area seen by the agent, C_a the area reachable.*

Table I: Parameter values tested and chosen.

Param	**Values**	**MF**	**MB**
α	0.0001, 0.001, 0.01, 0.1, 0.5, 0.9	0.5	0.01
γ	0.1, 0.5, 0.98, 0.9999	0.9999	0.98
τ	0.01, 0.1, 0.5, 1.0, 5.0	0.1	0.01

Blocks are carried on a conveyor belt, being spaced regularly and moved at **B**lock **S**peed (BS).

Two cases are studied :

1) Regular case (RC): BS is kept constant during the whole experiment.
2) Speed Shift case (SS): after a certain time with constant speed, the BS changes.

The input state s is a 2×8 vector, constructed as follows: for $i \in [0, 7]$, $s_i(t) = 1$ if a block has been seen at $t-i$ and 0 otherwise, and for $i \in [8, 15]$, $s_i(t) = 1$ if a block has been seen, and 0 otherwise. In s, the agent has to decide between 3 actions. Each one has an intrinsic cost, and can modify environment and perception (see table II).

The parameters of each Expert controlling the agent alone are optimised over reward accumulation as a priority and then variance. α is the learning rate i.e. how fast new knowledge is integrated, γ is the discount factor and weights the influence of distant rewards, τ is the decision temperature and balance the exploration/exploitation trade-off. The values tested are shown in table I.

B. Performance and Expert selection

The behavior of the agent is evaluated in performance by the received Cumulative Reward (fig. 4). In each case, we compare the performance of an agent controlled only by one Expert or by both Experts with arbitration. In both cases, the MB alone performs worse than the MF alone, and the random criterion allows a better performance in mean than Experts alone. It differs from [11] where the random criterion performance in the Regular Case corresponded to the mean between the performance of each Expert alone. This can be explained by the *a posteriori* arbitration of this architecture, whereas it was an *a priori* arbitration based only on the reception of a new State in the previous architecture. The *a posteriori* arbitration allows to ensure that each Expert has proposed an action for the current state before arbitrating whereas it may happen before the Experts decided when done *a priori*. Moreover, this set of parameters – different from [11] – leads to the MF alone having a bimodal performance distribution (fig. 5). Several policies can bring reward leading to the learning of several action sequences. In the worst experiments, the MF can be stuck in the starting state, or follow a low performance policy (e.g. using LC in states where it could use DN). The interaction between MB and MF benefits to the whole system by making a less varying performance, leading to a much more unimodal distribution.

Table II: Actions.

Action	Description	Cost c
Do nothing (DN)	Waiting action: no environmental modification nor perceptual information.	0
Look Cam (LC)	No environmental modification but updates p^{bs}.	-0.03.
Push Arm (PA)	This action removes a block being in C_a (thus providing a reward $R_t = 1$) and updates p^{bt}.	-0.03.

In the Regular Case, the Entropy criterion is equivalent in mean to using only the MF Expert whereas the Mean Reward criterion is close to the MB Expert alone performance. It is consistent with the mean selection rate of each Expert (fig. 6) and the bias induced by each criterion : the Entropy criterion (on the second row) shows that MF is preferentially selected. Because of our asynchronous implementation and the incremental learning of new states (see fig. 7), the Transition Model grows in complexity and the planification cannot be completely done in the given time. Thus, the MB Expert always keeps an overall higher entropy than the MF Expert. Further discussion and some improvement options will be exposed in Sect. IV.

The mean reward criterion tends to select the MB in most situations (fig. 6), since in this task setup, a negative feedback (i.e. a cost) is received when doing an action without getting reward. As the performance of this Expert is lower than the MF Expert one, it leads to a low performance of the Expert combination arbitrated with this criterion.

Besides, none of the four fusion methods did perform better than Experts alone nor than the Random criterion in the Regular Case. Both Experts contribute equally to the final action distribution so the uncertainty of the MB balances the higher certainty of the MF, and the final policy is worse than the one the MF alone is able to learn. This again confirms that the Regular Case is suitable for

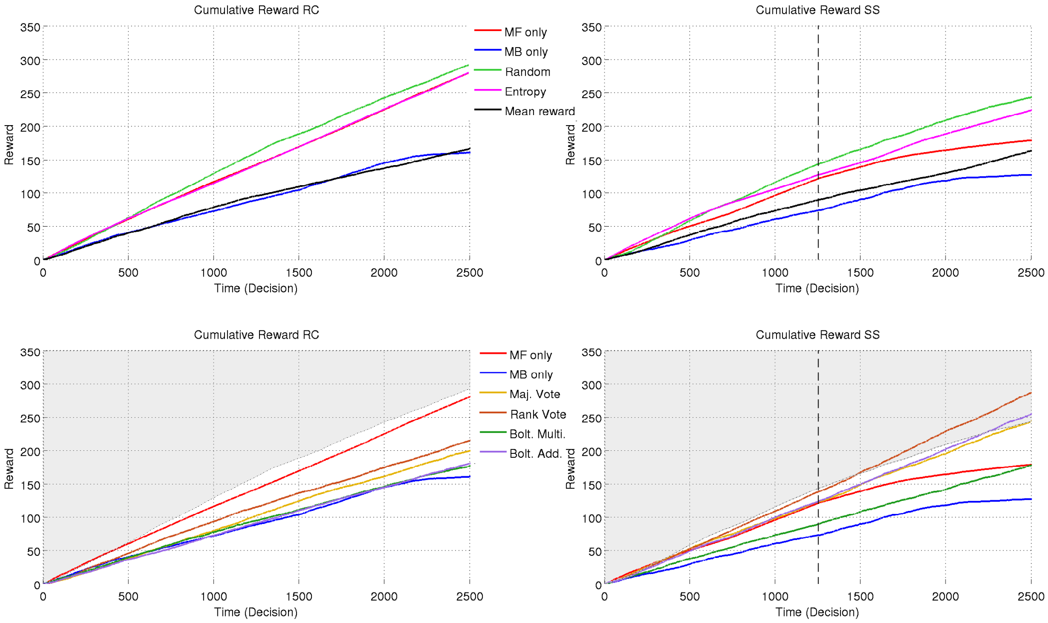


Figure 4: *Average Cumulative Reward received by the agent over 10 runs of the Block Push experiment. RC: regular case; SS: speed-shift case. First row: the signals-based criteria. Second row: the fusion-based ones. Dashed line time of the speed shift. The gray area is the zone where the cumulative reward is higher than the one obtained with the Random criterion, which is used as a reference.*

learning a behavioral ”habit” with the MF Expert whose decisions should not be polluted by the MB expert.

In the Speed Shift case however, the performance of Experts alone are strongly altered by the speed shift, leading to less rewarding policies (fig. 4). However, all combination methods (signal or fusion) lead to a performance robust to this environmental change, catching up the final MF alone performance (or being close for the Mean reward).

Three out of four fusion-based methods lead to a final performance higher than or equivalent to the Random performance. The very close performances of Majority Voting and Boltzmann Addition are surprising: Boltzmann Addition is more similar to Rank Voting in its principle and was expected to have similar performance. Majority Voting was expected to perform close or better than random selection : with only two Experts, arbitration when they disagree is equivalent to choosing action randomly, when they agree, the distribution is strongly peaked on the corresponding action, leading to a higher probability of selection. On the other hand, Boltzmann Addition keeps fine variations: summing probabilities may make two actions equiprobables whereas they were well-separated from each Expert’s point of view. The same phenomenon happens with Majority Voting : agreement leads to high certainty on the action to do, disagreement fades each Expert certainty in the collective decision. In addition, authors in [13] show a better performance of Boltzmann Multiplication and Majority voting, whereas here, the best method is Rank Voting. This can be explained by the fact that this method reinforce small constrasts in the probability distribution : even when two actions have very close probabilities, they are separated by the same score distance than the one between the first best action and a second best action with a far lower probability. Thus, with two Experts, small differences are highlighted and help quickly taking into account the few feedbacks received at the beginning. Finally, the Boltzmann Multiplication performs the worst among fusion-based methods. It suffers from directly using MB probabilities and thus fading the peaked distributions from MF (whereas Rank Voting and Boltzmann Addition keep the variations).

IV. Discussion

In this work, we evaluated the use of two signals to arbitrate between the MF and MB Experts. We highlighted that this approach only requires information that is already present in each Expert running alone,

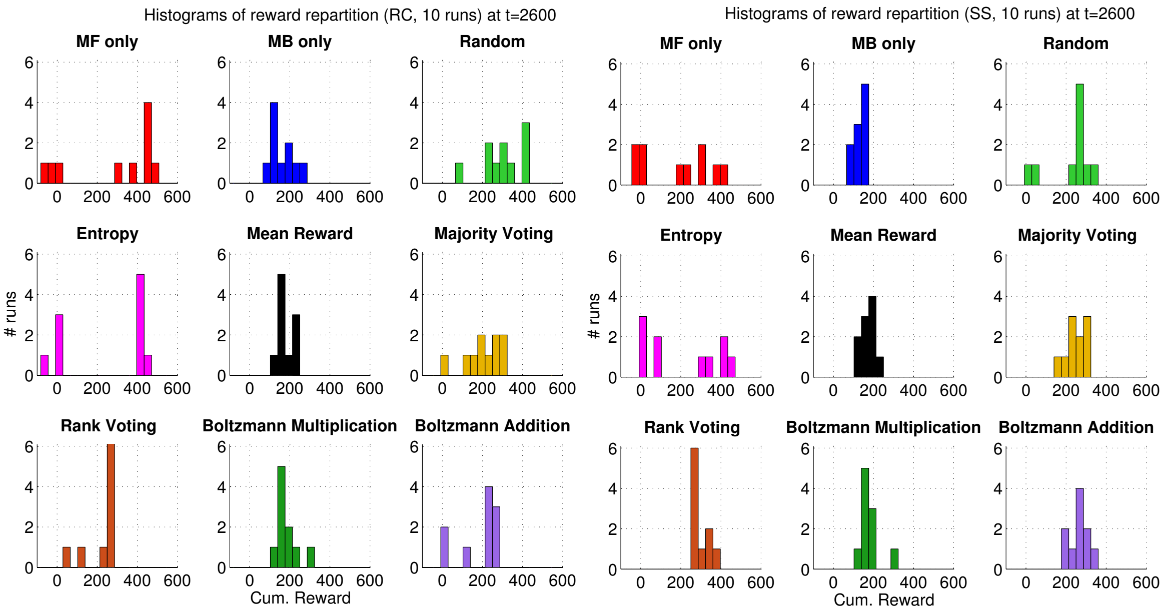


Figure 5: *Histograms of 10 runs of the Block Push experiment per criterion. Left : Regular Case (RC). Right : Speed Shift case (SS). For each case and each criterion, we plot the distribution of cumulative reward reached at time t. It shows that the performance can either be unimodal or bimodal.*

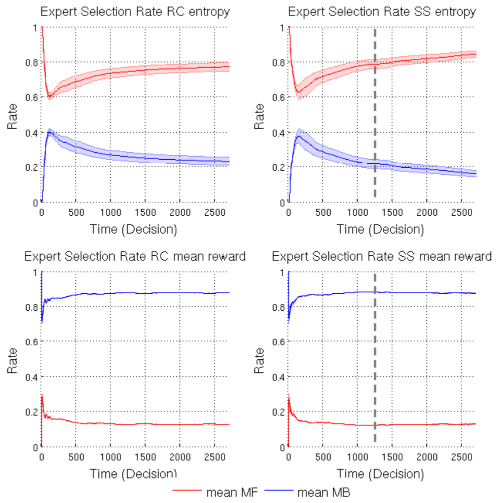


Figure 6: *Mean selection rate over 10 runs of the Block Push experiment per criterion. First row: Entropy arbitration. Second row: Mean reward arbitration. Left column: Regular Case (RC). Right column: Speed Shift Case (SS). The dashed line indicates the speed shift.*

and is related to environmental changes and Experts' certainty about decision. We also widened our interest to another approach that merges Experts' expertise, letting the Meta-Controller take the final decision from a unified probability distribution. We showed that, in the Regular Case condition, none of these arbitration and fusion methods clearly outperforms the random arbitration method used as a proof of concept in [11]. However, they allow to keep efficient policies when the agent is confronted to a change of environmental condition and thus a more robust behavior.

When signal-based arbitrations were tested, the selection rate of Experts showed a high bias from the criterion design. If Mean reward indicates inadequation of policy, it cannot predict when to give control to the MF Expert. The parametrization is also very dependent of the task: a high α value will make the agent very sensible to small changes in its policy and actions resulting from an exploratory choice in the softmax decision process will make the arbitration switches between Experts. A low value will make the agent blind to some changes that may require a shift back to the MB Expert. Thus,

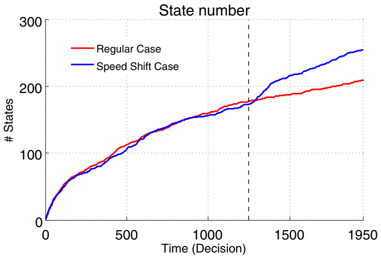


Figure 7: *Evolution of the number of states over time.*

tuning correctly this parameter would require to analyse the dynamics of the environment.

The Entropy criterion proposed here is close to the idea of uncertainty presented in [7]. In our high-dimensional task (fig. 7), the MB Entropy stays high, which biases the arbitration between Expert. Because our task has a dynamic environment, probabilistic transitions, the hypothesis of Perfect Information made in [8] is not valid, as it considers deterministic, synchronous and low dimensionality tasks. This point questions the direct application of RL algorithms to robotics : a RL agent performs efficiently if it can learn with certainty the relevant policy, or State-Action association. If the mapping of states in the task environment is ambiguous, learning the relevant policy in the real world will be difficult and far from optimality. Some recent work started to address the question of building relevant abstract representations from perception [15][16], and will be a key advance to make discrete RL algorithms efficient in real world situations. Also, because of the task's properties, it is tedious for Experts to learn a close-to-optimal policy. Thus, the relevance of criteria must be studied on other problems in order to validate or invalidate their interest for robotics, which will be the topic of further investigation.

These limitations also question the assumption made in Neuroscience models that performing a goal-directed behavior with an MB expert systematically implies to replan from scratch before each decision (thus explaining high reaction times in this situation; [7][8]). This assumption is reasonable only when confronted to very simple tasks with a small number of states. Our main limitation in performance here comes from the MB Expert, because of its bounded planning on a large number of states. An alternative approach consists in letting enough time for the planning to refine the action values, and then rely on this computed plan for next decisions, as is often done in robotics, where a plan is entirely computed before acting [17]. As our agent incrementally discovers its environment, this replanning should be done regularly, when needed, and entirely to exploit the new knowledge. In such a different setup, and taking inspiration from the fusion-based methods, do Experts have to propose an action or only the processed information for the final decision? The second option will allow to separate the planning process and the decision, which are coupled in our architecture, but do not have the same time scale. The decision should be made anytime a new state is received to allow reactivity, whereas the planning should be done only when our model of the task is too different than what is currently experienced by the agent.

Acknowledgements

This work has been funded by a DGA (French National Defence Agency) scholarship (ER), by the Project HABOT from Ville de Paris and by French Agence Nationale de la Recherche ROBOERGOSUM project under reference ANR-12-CORD-0030.

References

[1] A. Dickinson, "Actions and habits: The development of behavioral autonomy," *Philos. Trans. R. Soc. (London)*, vol. 308, pp. 67 – 78, 1985.

[2] R. J. Dolan and P. Dayan, "Goals and habits in the brain.," *Neuron*, vol. 80, pp. 312–325, Oct. 2013.

[3] B. W. Balleine and A. Dickinson, "Goal-directed instrumental action: contingency and incentive learning and their cortical substrates.," *Neuropharmacology*, vol. 37, pp. 407–419, 1998.

[4] H. H. Yin, S. B. Ostlund, and B. W. Balleine, "Reward-guided learning beyond dopamine in the nucleus accumbens: the integrative functions of cortico-basal ganglia networks.," *Eur. J. Neurosci.*, vol. 28, pp. 1437–1448, 2008.

[5] B. W. Balleine and J. P. O'Doherty, "Human and rodent homologies in action control: corticostriatal determinants of goal-directed and habitual action.," *Neuropsychopharmacology*, vol. 35, pp. 48–69, 2010.

[6] R. S. Sutton and A. G. Barto, *Introduction to Reinforcement Learning*. Cambridge, MA, USA: MIT Press, 1st ed., 1998.

[7] N. Daw, Y. Niv, and P. Dayan, "Uncertainty-based competition between prefrontal and dorsolateral striatal systems for behavioral control.," *Nat. Neurosci.*, vol. 8, no. 12, pp. 1704–1711, 2005.

[8] M. Keramati, A. Dezfouli, and P. Piray, "Speed/accuracy trade-off between the habitual and goal-directed processes.," *PLoS Comp. Biol.*, vol. 7, no. 5, pp. 1–25, 2011.

[9] L. Dollé, D. Sheynikhovich, B. Girard, R. Chavarriaga, and A. Guillot, "Path planning versus cue responding: a bioinspired model of switching between navigation strategies," *Biological Cybernetics*, vol. 103, no. 4, pp. 299–317, 2010.

[10] K. Caluwaerts, M. Staffa, S. N'Guyen, C. Grand, L. Dollé, A. Favre-Félix, B. Girard, and M. Khamassi, "A biologically inspired meta-control navigation system for the psikharpax rat robot," *Bioinspiration & Biomimetics*, vol. 7, p. 025009, 2012.

[11] E. Renaudo, B. Girard, R. Chatila, and M. Khamassi, "Design of a control architecture for habit learning in robots," in *Biomimetic and Biohybrid Systems, LNAI Proceedings*, pp. 249–260, 2014.

[12] E. Gat, "On three-layer architectures," in *Artificial Intelligence and Mobile Robots*, MIT Press, 1998.

[13] M. A. Wiering and H. van Hasselt, "Ensemble algorithms in reinforcement learning.," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 38, no. 4, pp. 930–936, 2008.

[14] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," in *ICRA Workshop on Open Source Software*, 2009.

[15] A. Droniou, S. Ivaldi, and O. Sigaud, "Deep unsupervised network for multimodal perception, representation and classification," *Robotics and Autonomous Systems*, 2014.

[16] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, pp. 529 – 533, 2015.

[17] R. Alami, R. Chatila, S. Fleury, M. Ghallab, and F. Ingrand, "An architecture for autonomy," *IJRR Journal*, vol. 17, pp. 315–337, 1998.