Autonomous selection of the “what” and the “how” of learning: an intrinsically motivated system tested with a two armed robot

Vieri G. Santucci
Ist. di Scienze e Tecnologie della Cognizione (ISTC)
Consiglio Nazionale delle Ricerche (CNR)
Via San Martino della Battaglia 44, 00185 Roma, Italia
School of Computing and Mathematics
University of Plymouth
Plymouth PL4 8AA, United Kingdom
vieri.santucci@istc.cnr.it

Gianluca Baldassarre and Marco Mirolli
Ist. di Scienze e Tecnologie della Cognizione (ISTC)
Consiglio Nazionale delle Ricerche (CNR)
Via San Martino della Battaglia 44, 00185 Roma, Italia
{marco.mirolli,gianluca.baldassarre}@istc.cnr.it

Abstract—**In our previous research we focused on the role of Intrinsically Motivated learning signals in driving the selection and learning of different skills. This work makes a further step towards more autonomous and versatile robots by implementing a 3-level hierarchical architecture that provides a system with the necessary mechanisms to both select goals to pursue and search for the best way to achieve them. In particular, we focus on the crucial importance of providing artificial agents with a decoupled architecture that separates the selection of goals from the selection of solutions. To verify our hypothesis, we use the architecture to control the two redundant arms of a simulated iCub robotic platform tested in a reaching task within a 3d environment. We compare its performance to the one of a system with a coupled architecture where the different goals are associated at design-time to different modules controlling the robot.**

I. INTRODUCTION

Developing artificial agents able to autonomously discover, select and solve new tasks is an important issue for robotics. This becomes even crucial if we want our robots to interact with real environments where agents have to face many unpredictable problems and where it is not clear which skills will be the more suitable to accomplish different goals.

Intrinsic Motivations (IMs) identify the ability of humans and other mammals (e.g, rats and monkeys) to modify their behaviour and learn new skills in the absence of a direct biological pressure. First studied in animal psychology (e.g. [1] [2]) and human psychology (e.g. [3] [4]), recently IMs have been investigated also with respect to their neural basis, with both experiments (e.g. [5] [6]) and computational models (e.g. [7] [8]).

IM learning signals can be considered a useful tool for the implementation of more autonomous and versatile robots, driving the formation of ample repertoires of skills without any assigned reward or task. In the last decades many computational researches based on IMs have been proposed (e.g. [9] [10] [11] [12] [13] [14]) and nowadays IMs are an important field of research also within robotics [15].

In particular, IMs can play an important role in guiding an artificial system to select its own goals: when many different skills can be acquired, it is crucial for the system to properly select only those that can be learnt and to focus on them only for the the time necessary to learn them. In previous work [16] we analysed which IM signal is more suitable to drive the selection and learning of multiple skills in a robotic system implemented with a hierarchical architecture. We compared different signals taken from the computational literature and we found that the best signals were based on the prediction error (PE), or prediction error improvement (PEI), of a predictor of the competence of the system in achieving the goals. These results underlined the role of goals in improving robotic learning processes [17] [18] [13] and the importance of using competence-based IM (CB-IMs) instead of knowledge-based IM (KB-IMs) learning signals to optimise the acquisition of a repertoire of skills (on the difference between CB-IMs and KB-IMs see [19] [20]).

In [16] we used a simple robotic setup, involving a 2 degrees-of-freedom (2DoF) robotic arm, tested in a 2D environment. Moreover, the architecture presented a significant limitation: a fixed coupling between the goals and the “experts” (modules), so that the system was forced to use a specific expert to learn a specific task.

Here we implement a more complex experimental setup, using the two redundant arms of a simulated iCub robotic platform tested in a reaching experiment within a 3d environment. We then focus on tackling the limit of our previous architecture, implementing the same CB-IM mechanism identified in [16] in a 3-levels hierarchical architecture that guarantees a decoupling between selected goals and experts. The system is so able not only to autonomously choose its own goals but also to autonomously determine with which expert (and hence effector) trying to achieve it and learn the related skill.

In particular, we focus on the importance of such a decoupled architecture to enhance the flexibility of artificial systems

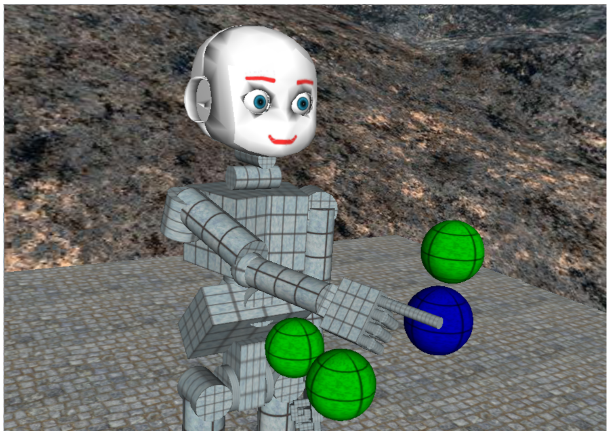


Fig. 1. The experimental setup, with the simulated iCub and the 4 objects. The green objects are those that the decoupled system learns to reach with the left arm, the blue object is reached with the right arm

and to improve their ability to autonomously discover suitable solutions to different problems. To verify our hypothesis, we compare the new system to one with fixed connections between goals and experts showing and analysing their performances in a reaching task where it is not clear which is the most suitable arm to reach for the different objects.

II. Setup

A. The simulated robot and the experimental setup

The robot is a reproduction of the iCub robotic platform, implemented with the FARSA simulator [21] developed in our institute (http://laral.istc.cnr.it/farsa). In the experiments presented here we only use the two redundant arms of the robot with 4DoF (the joints of the wrist and those of the fingers are kept fixed) in kinematic modality, so that collisions (that are not necessary for this test) are not taken into considerations. The fingers of the two hands are all closed with the exception of the two forefingers that are kept straight.

The task (Fig. 1) consists in learning to reach with the fingertip of the forefingers to the 4 fixed spherical objects (with radius set to 0.04 metres) positioned in the workspace of the two arms of the robot. Since we want to test the importance for an artificial system to autonomously search for the best solutions to the goals, the objects are all close to the Y axis that divides the workspace of the arms in right and left. The objects are all reachable using both the two hands of the robot, however it is not evident a priori which is the best solution (i.e. which arm to use to reduce the time spent in learning the task) to reach for each different object.

Note that this is just a simple example of a more general problem that real robots have to face in real environments: the impossibility of determining at design-time which will be the best strategy to interact with the world.

B. Architecture and coding

Since we want the robot to learn different skills and store them in its repertoire of actions, we use a hierarchical architecture where different abilities are stored in different components (the experts) of the system [22]. In our previous work, the system presented a 2-levels hierarchical architecture, with a goal selector determining on which goal the robot focused on each trial and different experts learning and storing the different skills. However, in that architecture the experts were coupled with the different goals at design-time, so that selecting a goal determined also with which expert the system tried to achieve it. This was a great limitation since a truly autonomous agent has to be able to select not only its goals but also how to achieve them. This is crucial because it is not possible to establish a priori the expert that is the proper one to learn a specific skill. For example, in the task presented here, it is not possible to determine which is the best arm to reach an object only on the basis of its position. In this sense, we define as a “coupled system” (CS) an architecture that, similarly to our previous work, has fixed connections between goals and experts used to achieve them, while we define as a “decoupled system” (DS) an architecture that is able to autonomously select both its goals and how to accomplish them (i.e. the expert controlling the robot effectors).

To verify the importance of such a decoupled architecture to foster the autonomy and flexibility of artificial agents, in the present work we implement a DS with 3-levels (Fig. 2): 1) a high-level selector that determines which goal to pursue (here the object that the robot is trying to reach); 2) a low-level selector that determines which expert controls the robot, hence the arm used to reach the goal and learn the related skill; 3) a control layer of n experts, half controlling the right arm half controlling the left arm.

The goal selector is composed by 4 units, one for each possible goal (the 4 spheres). At the beginning of every trial, it determines through a winner-takes-all (WTA) *softmax* selection rule [23] which goal to pursue. The probability of unit k to be selected (p_k) is thus:

$$p_k = \frac{exp\frac{Q_k}{\tau}}{\sum_{i=0}^{n} exp\frac{Q_n}{\tau}} \tag{1}$$

where Q_k is the value of unit k and τ is the *temperature* value, set to 0.008, which regulates the stocasticity of the selection. The activation of each unit is determined by an exponential moving average (EMA, with a smoothing factor set to 0.35) of the intrinsic reinforcement obtained for pursuing that goal (for the description of the CB-IM mechanism generating the IM reinforcement signal, see Sec. II-C).

The selector of the experts is formed by n units, one for each expert, fully connected with the units of the goal selector. At the beginning of every trial it receives as input the information on which goal has been selected by the goal selector (encoded in a 4-elements binary vector) and determines the expert (and hence the arm) controlled by the system during the trial through a WTA *softmax* selection rule (see Eq. 1) with *temperature* set to 0.05. The activity of each unit is determined by the weight connecting that unit with the one of the selected goal. At each trial, the weight is updated through an EMA (with smoothing factor set to 0.35) of the reward obtained to achieve the selected goal (1 for success, 0 otherwise).

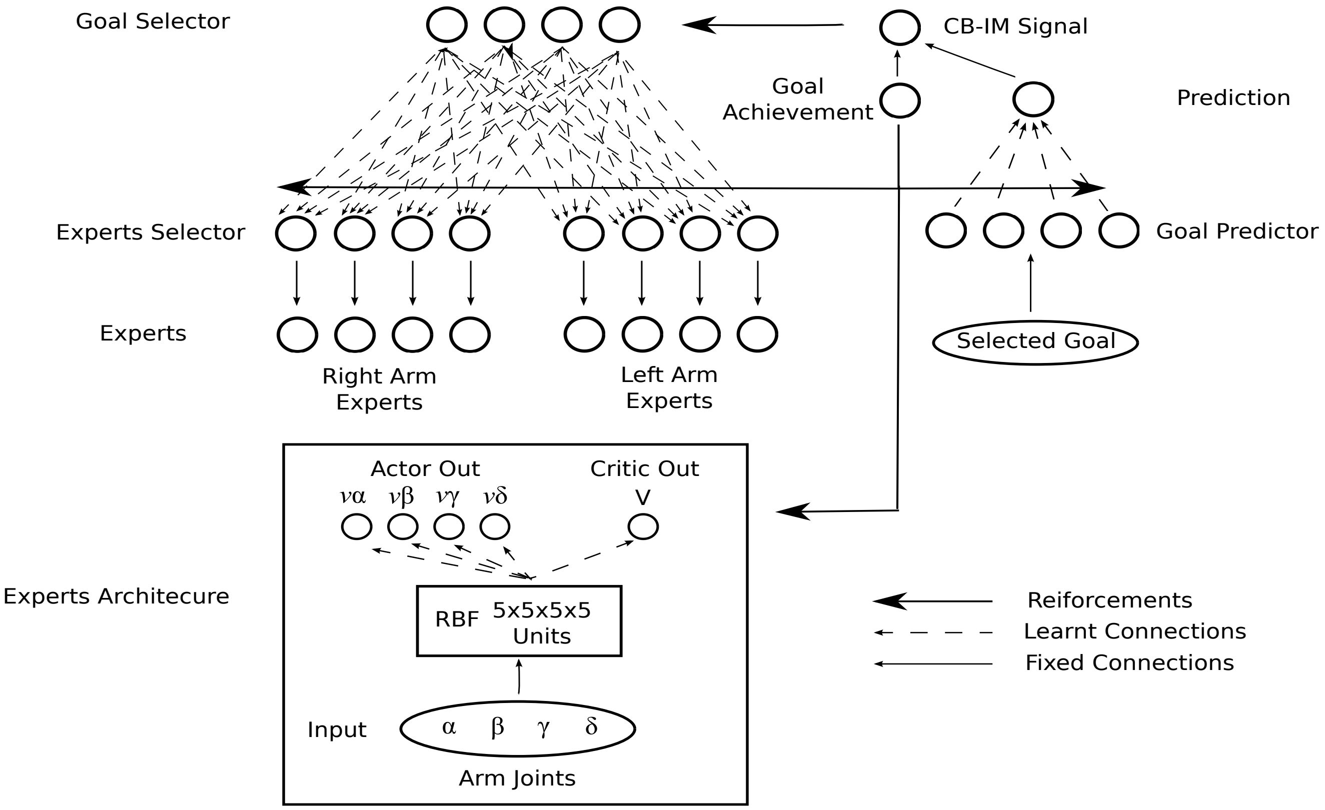


Fig. 2. The 3-level hierarchical architecture implemented to control the robot together with the mechanism (the predictor) determining the CB-IM signal. For a detailed description see Sec. II-B

Each expert is a neural network implementation of the actor-critic architecture [24] adapted to work with continuous state and action spaces [25]. The input to each expert consists in the 4 actuated joints of the related arm (3 joints for the shoulder, 1 for the elbow), α β γ δ (all within the ranges of the real robot), coded through Gaussian radial basis functions (RBF) [26] in a 4 dimensional grid with 5 units per dimension.

The evaluation of the critic (V) of each expert is a linear combination of the weighted sum of the input units plus a bias unit with fixed input set to 1. The actor of each expert has 4 output units, fully connected with the input, with a logistic transfer function:

$$o_j = \Phi\left(b_j + \sum_{i}^{N} w_{ji} a_i\right) \qquad \Phi(x) = \frac{1}{1+e^{-x}} \tag{2}$$

where b_j is the bias of output unit j, N is the number of input units, a_i is the activation of input unit i and w_{ji} is the weight of the connection linking unit i to unit j. Each motor command o_j^m is determined by adding noise to the activation of the relative output unit j (o_j). Since the controller of the robot modifies the velocity of the joints progressively, a simple random noise would turn out to determine extremely little movements. For this reason, similarly to [25], we generate the noise value (nv) with a normal Gaussian distribution with average 0 and standard deviation (SD) 2.0 and pass it through an EMA with a smoothing factor set to 0.08.

To reduce the time spent by the experts to reach the targets when their competence improves, we implemented an algorithm to let the system self-modulate the generated nv, changing the SD for each expert with a “noise-decrease value” (ndv) determined by an EMA (with smoothing factor set to 0.0005) of the success of the expert in reaching the targets (1 for success, 0 otherwise). More precisely, the SD for expert e at time t (SD_{et}) is calculated as follow:

$$SD_{et} = SD(1 - ndv) \tag{3}$$

The actual motor commands are then generated as follows:

$$o_j^m = o_j + nv \tag{4}$$

where the resulting commands are limited in [0; 1] and then remapped to the velocity range of the respective joints of the robot determining the applied velocity ($v\alpha$, $v\beta$ etc.).

The experts are trained through a TD reinforcement learning algorithm. The TD-error of expert e ($TDerr_e$) is computed as:

$$TDerr_e = (R_e^t + \gamma_k V_e^t) - V_e^{t-1} \tag{5}$$

where R_e^t is the reinforcement for the expert at time step t, V_e^t is the evaluation of the critic at time step t, and γ is a discount factor set to 0.99. The reinforcement is 1 when the robot touches the selected target, 0 otherwise. The connection weight w_i of critic input unit i is updated as usual [23]:

$$\Delta w_i = \eta^c \delta a_i \tag{6}$$

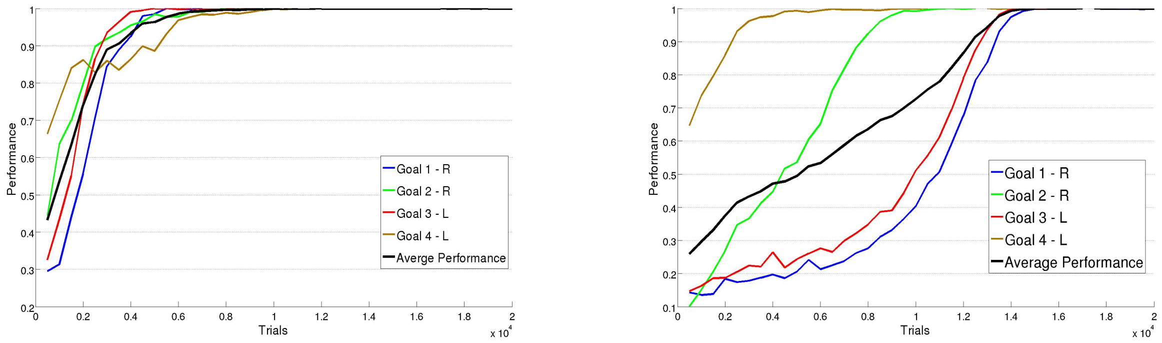


Fig. 3. Performance on reaching the different objects (Goal 1, 2, 3 and 4. The label R means the related object is positioned on the right with respect to the Y axis dividing the workspace; label L means the object is positioned on the left) and average performance on all the objects (Average Performance) of the DS (left) and the CS (right).

where η^c is a learning rate, set to 0.02. The weights of each actor are updated as follows [27]:

$$\Delta w_{ji} = \eta^a \delta (o_j^m - o_j)(o_j(1 - o_j))a_i \tag{7}$$

where η^a is the learning rate, set to 0.4, $o_j^m - o_j$ is the difference between the action executed by the system (determined by adding noise) and that produced by the controller, and $o_j(1 - o_j)$ is the derivative of the logistic function.

C. CB-IM mechanism

The reinforcement signal (R_s^t) driving the selection of the goals is the intrinsic reinforcement generated by the CB-IM mechanism we identified in [16] as the best suitable to drive the selection of different goals and the acquisition of the related skills. In particular, R_s^t is the prediction error improvement (PEI) of a predictor that receives the selected goal as input (encoded in a 4-elements binary vector, with 4 being the number of the goals) and produces a prediction in the range [0, 1] on the achievement (within the time-out of the trial) of the selected goal. At time t, the PEI is calculated as the difference between the average absolute prediction errors (PEs) calculated over a period T of 40 trials:

$$PEI_t = \frac{\sum_{i=t-(2T-1)}^{t-T} |PE|_i}{T} - \frac{\sum_{i=t-(T-1)}^{t} |PE|_i}{T} \tag{8}$$

The predictor is trained through a standard delta rule using the achievement of the selected goal as teaching input (1 for success, 0 otherwise) and with a learning rate set to 0.05.

D. Compared systems and experimental settings

To test the importance for an artificial system to autonomously select and learn how to achieve different goals, we compare the presented system to one with an architecture similar to [16], where there was no decoupling between the experts and the goals. In such a CS the first and second level of our new architecture are flattened in a single layer, so that the unique selector selects an expert to which is directly associated a goal (the object to be touched). All the other elements, mechanisms and parameters are identical for both architectures except for the number of experts.

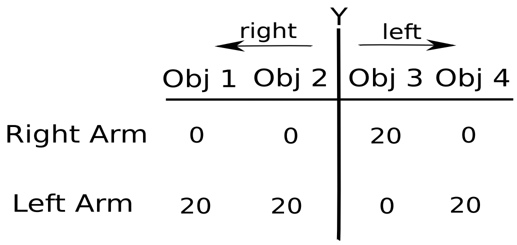
Since it is possible that the best solution is to reach for every object with the same arm, the decoupled system (DS) has 8 experts, 4 controlling each arm, so that is potentially able to learn to reach every object with a different expert of the same arm. Differently, the coupled system (CS) has only 4 experts, 2 for each arm: we associate the spheres on the right side of Y axis with the experts controlling the right arm (1 each) and those on the left side with the 2 experts controlling the left arm (1 each).

The experiment lasts 20,000 trials. At the beginning of every trial the goal selector determines which of the 4 spheres is the target. Then, in the DS the selector of the experts determines which expert (and hence which arm) will be used to learn to reach for that object, while in the CS the control goes to the expert (and to the arm) associated at design-time to that object. The joints of the selected arm are then randomly initialised. The trial ends when the selected goal is achieved (the robot touches the selected object) or after a time out of 800 time steps, each lasting 0.05 seconds.

III. Results

The performance of the two systems in the reaching task is shown in Fig. 3 (data show the average performance of 20 replications of each experiment). As in [16] the CB-IM signal is able to drive the systems to learn all the skills related to the different goals. However, the DS learns significantly faster than the CS. If we look at the single tasks we can see that while the DS is able to learn to reach all the 4 objects very quickly, the CS is able to rapidly learn to reach object 4 (even faster, on average, than CS, that first focuses on the other objects) while it takes more time to achieve an high performance on the other goals, especially number 1 and 3. If we analyse the results of the DS it is clear why this system performs better.

Fig. 4 summarises the solutions adopted by the DS to reach the 4 objects in the different replications of the experiment. In 3 cases (objects 1, 2 and 3) the system learns to reach the target with the opposite arm with respect to the position of the object on the Y axis (see also Fig. 1). Those 3 cases are the goals where the CS is slower than the DS. While our new system has an architecture that is able to autonomously search for the


Y
right
left

	Obj 1	Obj 2	Obj 3	Obj 4
Right Arm	0	0	20	0
Left Arm	20	20	0	20

Fig. 4. Summary of the solutions adopted by the DS to reach the different objects, with respect to the position of the objects and the arm used to reach for it in the 20 replications of the experiment.

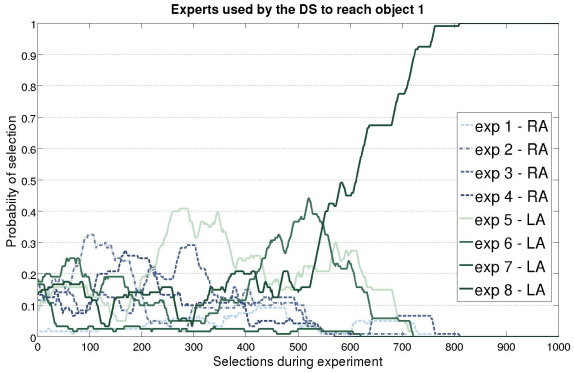


Fig. 5. Experts selection, with respect to the control of right arm (RA) and left arm (LA), for the achievement of goal 1 in a representative replication of the experiment with the DS. Data are related to the first 1,000 selections of that goal. After them the system has learnt to systematically associate a specific expert (exp 8 - LA) to the goal.

best solution to achieve the different goals, the CS is forced, by definition, to use the expert (and then the arm) associated with an object at design-time when it is extremely difficult (or even impossible, if we imagine more complex tasks) to determine the most suitable strategy to learn each skill.

The DS instead is able to test the different experts and find the solution that guarantees a better performance. In Fig. 5 we show the history of experts selections related to goal 1 in a representative replication of the experiment with the DS. At the beginning, the system tries to achieve the goal with different experts controlling both the arms but, after some time, the system learns to achieve that goal by using always one of the experts controlling the left arm. Note that, in principle, a decoupled architecture may suffer the problem of catastrophic interference [28] if it is not able to assign different experts to different skills: however, this does not happen in our system, which is able to efficiently learn to reach each object through a different expert (on this issue see also [29] [30] [31]).

IV. Conclusion

In this work we implemented a 3-level hierarchical architecture controlling the redundant arms of a simulated iCub robotic platform and we tested the importance to autonomously select the resources (the experts) to discover the best solutions to achieve its autonomously-selected goals. To drive the autonomous selection of goals, we used Intrinsic Motivations (IMs) implemented through the mechanism generating the CB-IM reinforcement signal that we identified in our previous research [16]. We provided the system with an architecture that allows the robot to autonomously select both its goal and the expert (hence the arm) to achieve it. We built an experimental setup consisting in a reaching tasks with 4 objects in a 3D environment and we compared the implemented decoupled system (DS) with a coupled system (CS) that has fixed connection between goals and experts.

The results show that our autonomous system is able to select and learn the different skills. Moreover, the experiments show that the DS performs significantly better than the CS. The reason of these results lies in the different structure of the architectures of the two systems: the DS is able to discover the best solutions to reach for the different objects while the CS is forced to use the experts (and then the arm) associated to each goal at design-time.

This is just a simple test to show a crucial issue for real robots that have to act in complex environments: when there are many different goals that can be achieved, it is not possible to determine a priori which are the best strategies to solve all the problems the robot will have to face. Improving the ability of an artificial agent not only in selecting its own goals but also in searching for the best solutions to reach them is a necessary step towards more flexible and autonomous robots. The architecture we presented in this work is able to guarantee this two-level autonomy, supporting the system in exploring different goals and finding the appropriate strategies to achieve them faster.

In future works we will test the robot with more difficult tasks and we will provide a wider range of different experts to the system. Here the robot can only choose to control one of the two arms, while a real agent can have more effectors to interact with the world. Moreover, the experts can vary also for their $inputs$ and for their internal $structure$, providing in this way different solutions also with the same $effector$. We showed that a system endowed with our architecture is able to autonomously select the resources (experts) to search and learn the best strategies to achieve different goals: our hypothesis is that the advantages of such an architecture will be better enlightened if the system is tested in more complex experimental setup or if the system has a wider range of different computational resources to accomplish its goals. Obviously, a simpler scenario where the best strategy is evident at design-time will advantage a coupled system like CS, since it does not have to waste time in searching for the best modules to train its skills. However, such a scenario is far away from the difficulties that a robot have to face in real-world environment, where autonomy and versatility are crucial to the succeed.

Moreover, in future works we will tackle a limit that still affects our architecture: the goals that the system can set are given at the beginning of the experiment. A further step towards more versatile agents is to provide the systems with the ability to autonomously discover new goals. Some efforts have been made in this direction in the field of hierarchical reinforcement learning but most of them (e.g. [32] [33]) focus on searching sub-goals on the basis of externally given tasks (reward function). Only few works (e.g. [34] [35] [13]) try to implement systems able to set their own goals independently from any specific task, which is the crucial condition to move towards a real open-ended autonomous development.

Acknowledgment

The authors want to thank Tommasino Ferrauto and Gianluca Massera for their precious help with the FARSA simulator.

This research has received funds from the European Commission under the 7th Framework Programme (FP7/2007-2013), ICT Challenge 2 "Cognitive Systems and Robotics", project "IM-CLeVeR - Intrinsically Motivated Cumulative Learning Versatile Robots", grant agreement no. ICT-IP-231722.

References

[1] H. F. Harlow, "Learning and satiation of response in intrinsically motivated complex puzzle performance by monkeys," *Journal of Comparative and Physiological Psychology*, vol. 43, pp. 289–294, 1950.

[2] R. White, "Motivation reconsidered: the concept of competence." *Psychological Review*, vol. 66, pp. 297–333, 1959.

[3] D. Berlyne, *Conflict, Arousal and Curiosity*. McGraw Hill, New York, 1960.

[4] R. M. Ryan and E. L. Deci, "Intrinsic and extrinsic motivations: Classic definitions and new directions." *Contemporary Educational Psychology*, vol. 25, no. 1, pp. 54–67, 2000.

[5] B. Wittmann, N. Daw, B. Seymour, and R. Dolan, "Striatal activity underlies novelty-based choice in humans," *Neuron*, vol. 58, no. 6, pp. 967–73, 2008.

[6] E. Duzel, N. Bunzeck, M. Guitart-Masip, and S. Duzel, "Novelty-related motivation of anticipation and exploration by dopamine (nomad): implications for healthy aging," *Neuroscience Biobehavioural Review*, vol. 34, no. 5, pp. 660–669, 2010.

[7] S. Kakade and P. Dayan, "Dopamine: generalization and bonuses." *Neural Networks*, vol. 15, no. 4-6, pp. 549–559, 2002.

[8] M. Mirolli, V. G. Santucci, and G. Baldassarre, "Phasic dopamine as a prediction error of intrinsic and extrinsic reinforcements driving both action acquisition and reward maximization: A simulated robotic study," *Neural Networks*, vol. 39, no. 0, pp. 40 – 51, 2013.

[9] J. Schmidhuber, "Curious model-building control system," in *Proceedings of International Joint Conference on Neural Networks*, vol. 2. IEEE, Singapore, 1991, pp. 1458–1463.

[10] A. Barto, S. Singh, and N. Chantanez, "Intrinsically motivated learning of hierarchical collections of skills," in *Proceedings of the Third International Conference on Developmental Learning (ICDL)*, 2004, pp. 112–119.

[11] M. Schembri, M. Mirolli, and G. Baldassarre, "Evolving internal reinforcers for an intrinsically motivated reinforcement-learning robot," in *Proceedings of the 6th International Conference on Development and Learning*, Y. Demiris, D. Mareschal, B. Scassellati, and J. Weng, Eds. Imperial College, London, 2007, pp. E1–6.

[12] P. Oudeyer, F. Kaplan, and V. Hafner, "Intrinsic motivation system for autonomous mental development," in *IEEE Transactions on Evolutionary Computation*, 2007, pp. 703–713.

[13] A. Baranes and P.-Y. Oudeyer, "Active learning of inverse models with intrinsically motivated goal exploration in robots," *Robotics and Autonomous Systems*, vol. 61, no. 1, pp. 49 – 73, 2013.

[14] V. Santucci, G. Baldassarre, and M. Mirolli, "Cumulative learning through intrinsic reinforcements," in *Evolution, Complexity and Artificial Life*, S. Cagnoni, M. Mirolli, and M. Villani, Eds. Berlin: Springer-Verlag, 2014, pp. 107–122.

[15] G. Baldassarre and M. Mirolli, Eds., *Intrinsically Motivated Learning in Natural and Artificial Systems*. Berlin: Springer-Verlag, 2013a.

[16] V. G. Santucci, G. Baldassarre, and M. Mirolli, "Which is the best intrinsic motivation signal for learning multiple skills?" *Frontiers in Neurorobotics*, vol. 7, no. 22, 2013.

[17] M. Rolf, J. J. Steil, and M. Gienger, "Goal babbling permits direct learning of inverse kinematics," *Autonomous Mental Development, IEEE Transactions on*, vol. 2, no. 3, pp. 216–229, 2010.

[18] V. G. Santucci, G. Baldassarre, and M. Mirolli, "Intrinsic motivation mechanisms for competence acquisition," in *Development and Learning and Epigenetic Robotics (ICDL), 2012 IEEE International Conference on*, 2012, pp. 1–6.

[19] P.-Y. Oudeyer and F. Kaplan, "What is intrinsic motivation? a typology of computational approaches," *Front Neurorobot*, vol. 1, p. 6, 2007.

[20] M. Mirolli and G. Baldassarre, "Functions and mechanisms of intrinsic motivations: The knowledge vs. competence distinction," in *Intrinsically Motivated Learning in Natural and Artificial Systems*, G. Baldassarre and M. Mirolli, Eds. Berlin: Springer-Verlag, 2013.

[21] G. Massera, T. Ferrauto, O. Gigliotta, and S. Nolfi, "Farsa: An open software tool for embodied cognitive science," in *Advances in Artificial Life, ECAL*, vol. 12, 2013, pp. 538–545.

[22] G. Baldassarre and M. Mirolli, *Computational and Robotic Models of the Hierarchical Organization of Behavior*. Berlin: Springer, 2013.

[23] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, 1998.

[24] A. Barto, R. Sutton, and C. Anderson, "Neuron-like adaptive elements that can solve difficult learning control problems," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 13, pp. 834–846, 1983.

[25] K. Doya, "Reinforcement learning in continuous time and space." *Neural Computation*, vol. 12, no. 1, pp. 219–245, 2000.

[26] A. Pouget and L. H. Snyder, "Computational approaches to sensorimotor transformations." *Nature Neuroscience*, vol. 3 Suppl, pp. 1192–1198, 2000.

[27] M. Schembri, M. Mirolli, and G. Baldassarre, "Evolving childhood's length and learning parameters in an intrinsically motivated reinforcement learning robot," in *Proceedings of the Seventh International Conference on Epigenetic Robotics*, L. Berthouze, G. Dhristiopher, M. Littman, H. Kozima, and C. Balkenius, Eds. Lund University Cognitive Studies, Lund, 2007, pp. 141–148.

[28] M. McCloskey and N. J. Cohen, "Catastrophic interference in connectionist networks: The sequential learning problem," *The psychology of learning and motivation*, vol. 24, no. 109-165, p. 92, 1989.

[29] R. Nishimoto and J. Tani, "Development of hierarchical structures for actions and motor imagery: a constructivist view from synthetic neurorobotics study," *Psychological Research PRPF*, vol. 73, no. 4, pp. 545–558, 2009.

[30] D. Caligiore, M. Mirolli, D. Parisi, and G. Baldassarre, "A bioinspired hierarchical reinforcement learning architecture for modeling learning of multiple skills with continuous states and actions," in *Proceedings of the Tenth International Conference on Epigenetic Robotics*, vol. 149, 2010.

[31] P. Tommasino, D. Caligiore, M. Mirolli, and G. Baldassarre, "Reinforcement learning algorithms that assimilate and accommodate skills with multiple tasks," in *Development and Learning and Epigenetic Robotics (ICDL), 2012 IEEE International Conference on*. IEEE, 2012, pp. 1–8.

[32] A. McGovern and A. G. Barto, "Automatic discovery of subgoals in reinforcement learning using diverse density," in *Proceedings of the Eighteenth International Conference on Machine Learning*, ser. ICML '01. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001, pp. 361–368. [Online]. Available: http://dl.acm.org/citation.cfm?id=645530.655681

[33] G. Konidaris and A. Barto, "Skill discovery in continuous reinforcement learning domains using skill chaining," in *Advances in Neural Information Processing Systems 22 (NIPS '09)*, Y. Bengio, D. Schuurmans, J. Lafferty, C. Williams, and A. Culotta, Eds., 2009, pp. 1015–1023.

[34] J. Mugan and B. Kuipers, "Autonomously learning an action hierarchy using a learned qualitative state representation," in *Proceedings of the 21st international jont conference on Artifical intelligence*, ser. IJCAI'09. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2009, pp. 1175–1180.

[35] C. Vigorito and A. Barto, "Intrinsically motivated hierarchical skill learning in structured environments," *IEEE Transactions on Autonomous Mental Development*, vol. 2, no. 2, pp. 132–143, 2010.