

Learning and motivational couplings promote smarter behaviors of an animat in an unknown world

Mathias Quoy*, Philippe Laroque, Philippe Gaussier

Université de Cergy-Pontoise, ETIS, 6 av. du ponceau, 95014 Cergy-Pontoise Cedex, France

Received 30 August 2001; received in revised form 30 September 2001; accepted 30 November 2001

Abstract

This paper reports on statistical results of an animat behavior in an unknown environment using a cognitive map. We study the coupling effect between the motivation for eating and the one for drinking. Some smart behaviors are not caused by a sophisticated “intelligent” algorithm, but only through coupling of the motivations. Adding a learning rule on the links of the cognitive map allows to reinforce particular paths, and to forget others. This also leads to reinforce the same “smart” behaviors. © 2002 Elsevier Science B.V. All rights reserved.

Keywords: Planning; Cognitive map; Neural networks; Learning

1. Introduction

The cognitive map approach to planning is not new [2,5,11,19,28,37–40]. However, adding a learning procedure on the links of the cognitive map enables to reinforce particular paths, and to forget others, allowing for an adaptation to a changing perception of the environment, as it is gradually discovered by the animat.

As far as planning is concerned, most models use graphs giving how to go from one place to the other [2,5,11,15,19,20,24,28,30,37,38,40]. The differences between them lay in the construction of this graph, the way it is used in order to find the shortest path, how it works when the environment changes or when there are different simultaneous goals to be achieved. Other works use rule-based algorithms (hierarchical classification [17]) or “potential fields” [3], or even classical A^* algorithms. The classical functional approach

relies on the a priori definition of desired behaviors [13] or on a complete world modeling [41]. These classical programming techniques may allow to produce navigation models exhibiting behaviors similar to ours. To sum up, the navigation problem coupled with a planning mechanism may be solved by dedicated efficient techniques. We want to stress here that the navigation/planning part of our system is only a piece of a more complex system which should be able to recognize objects [26], to imitate [1,29], and finally to communicate. So, from the beginning, we have to integrate the fact that our final goal is more than “just” navigating and planning. In particular, we stress that it is important to take into account the dynamical aspects of internal and external couplings. These dynamical aspects are explored here as the coupling between the animat’s motivations (eat, drink, rest, etc.), and their interaction with the environment. So, we will study how this coupling affects the animats behavior. In the following, we first present the experiment, then our model and the results achieved. Finally, we conclude.

* Corresponding author.

E-mail address: quoy@ensea.fr (M. Quoy).

2. Experiment

In our experiments, the animat is “living” in an unknown environment with obstacles, “food” and “water” sources and a “nest” to rest. The animat is switching between two behaviors: exploring the environment and planning. The animat has three contradictory motivations: eating, drinking and resting. Each of these motivations is linked with a satisfaction level decreasing over time, and increasing when on the proper supply source. When the level is below some threshold, the animat triggers the planning behavior for reaching the needed source.

Most experiments have been carried out in a maze environment. This enables us to control the kind of paths used by the animat by varying the size of the arms of the maze. A corridor is 7 animat’s width wide. The animat needs approximately 50,000 steps for exploring the whole maze.

The animat only gets two informations. A visual input gives the existence of a particular landmark. A compass gives the azimuth of these landmarks relatively to the magnetic north (top of the environment in the computer simulation). Hence, a particular location is defined by the set of (landmark, azimuth). There is no Cartesian map of the environment, nor use of a square paving as in some other planning algorithms (Q-learning for instance).

In the right arm of the T-maze, we place a food source. In the left one, there are both food and water. In the bottom of the vertical arm, there is a nest, where the animat rests. It is the initial starting location. The need for going to rest increases twice as fast as the ones for drinking or eating.

3. Model

The cognitive map is inspired by works on the possible use of “Place Cells” of the rat [31]. Place Cells are particular neurons of the CA3 region of the rat’s hippocampus. These neurons are typical for a particular location in the environment (they only fire when the rat comes to the corresponding location). Hence, the various places of a room for instance are coded by Place Cells becoming active when the rat comes into its response field. Some recent works of spatial navigation include the simulation of some biologically

plausible neuronal structures, and in particular the use of “Place Cells” as they may exist in the hippocampus [6,7,12,16,22,35,36,40].

In the beginning, the animat has no knowledge about the visual forms it can encounter, except that there are some landmarks. The set of landmarks (together with their azimuth) the animat can see at a given moment determines a “known” location. This location is coded in a particular neuron on a probabilistic topological map (PTM). We call this neuron a Place Cell. Initially, the weights of the connections in this map are randomly generated, so any neuron can win. When a new configuration is coded on a given Place Cell, the weights of its neighbors are updated with a probability depending on the Euclidean distance between the weight vector of the winner and that of each neighbor. That allows for a neuron “near” the winner to be able to react to a situation similar to the newly learned situation. A configuration is considered as “new” when the activation of the winner comes under a given threshold. The new set of (landmark, azimuth) is then learned on a new Place Cell. In that way, the exploration of the environment leads to the creation and connection of a various number of Place Cells. Eq. (1) gives the activity of place cell P_i out of a given set of (landmark, azimuth).

$$\text{Act}_{P_i} = 1 - \frac{\sum_{k=1}^{N_i} V_{i,k} \cdot f(|\Theta_{i,k} - \theta_k|, v_k)}{\pi N_i}, \quad (1)$$

where N_i is the number of visible landmarks when the robot is at the learned place i (or cell P_i). $\Theta_{i,k}$ represents the learned value of the azimuth of landmark k from the learned place i . θ_k is the value of the same landmark azimuth for the *current robot location*. All angles are expressed in radians and measured from an absolute direction (the north for instance, given by a compass). $|\Theta_{i,k} - \theta_k|$ is computed modulo π , $V_{i,k}$ and v_k are set to 1 when the landmark k is seen from the learned location i and from the current robot location, respectively (0 otherwise). When the system exhibits landmark recognition problems, we can have $V_{i,k} = 1$ and $v_k = 0$ (landmark learned but not recognized). f is a nonlinear function solving these landmark visibility problems:

$$f(\theta, v_k) = \begin{cases} \theta & \text{if } v_k = 1, \\ \pi & \text{if } v_k = 0. \end{cases}$$

The error associated with one landmark azimuth is maximum when the landmark cannot be found ($f(\theta, 0) = \pi$). Before learning, all neurons' synaptic weights are set to a random value, so the neurons response is very low. When learning is triggered (for instance when the goal is discovered), the most activated neuron is selected to learn that place ($\Theta_{i,k} \leftarrow \theta_k$). Next, according to Eq. (1) the activation of that neuron will grow and tend to be maximum ($Act = 1$) when the robot moves in the direction of the learned location (the azimuths θ_k associated with the current location are close to the stored $\Theta_{i,k}$). By using a simple gradient technique, it is theoretically possible to reach the goal wherever the starting point of the robot is [34].

So, the exploration of the environment leads to the building of a various number of Place Cells. At this time, there is no cognitive map, but only a collection of Place Cells. We have shown that it is already sufficient for reaching a particular goal when each Place Cell is associated with a movement to perform [23]. When thinking of a real robot, the first problem is to reduce the information flow coming from the CCD camera. It may be done by a parallelizing visual information processing [21,32]. But the main reduction is due to the fact that a particular behavior will not be associated with a whole scene, but only with the combination of local snapshots taken around particular points (landmarks) [14,25]. Thus, a view will be reduced to a set of snapshots. These snapshots are taken around

points of high contrast change. We could also take other techniques such as comparing panoramic views by computing their displacement field [18].

For more complex tasks, when several goals have to be achieved or when the system has to learn to navigate in a complex visual environment (the different goals do not belong to the same visual environment), the choice of the route to follow cannot only rely on simple sensory-motor associations. We tackle this problem by using a cognitive map. The map is built by linking together two Place Cells reached successively. This builds a graph which roughly represents the topology of the environment in Fig. 1.

This map alone is not enough for planning. It has to be associated with a motivational system so that the most interesting places (food and water sources) can be linked with the appropriate motivation. The activity of a neuron is now modulated by the motivation activity, so that we have:

$$Pot_{P_i} = Act_i \times Motiv_i, \tag{2}$$

where $Motiv_i$ is the motivation arriving at neuron i .

Now, when a particular need has to be fulfilled, the associated motivation triggers the activation of the appropriate neuron in the cognitive map. The diffusion of this activity along the graph until the actual location of the animat defines the route to follow for reaching the goal [10,11,15]. The neural diffusion we use is equivalent to the Bellman–Ford algorithm [8,34]. The activity of each neuron of the graph is the maximum

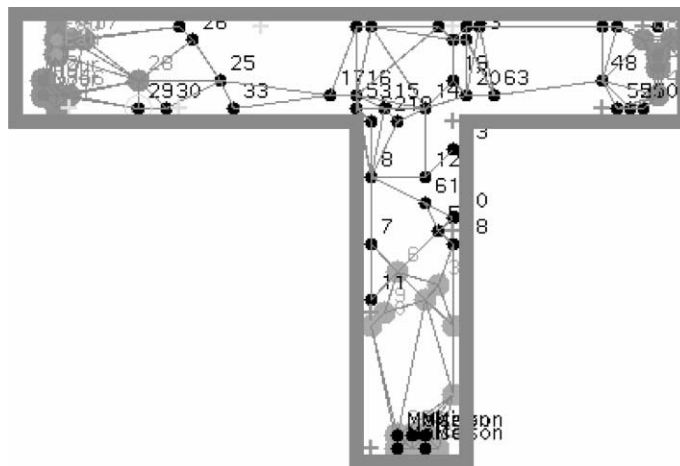


Fig. 1. The animat in a T-maze whose left arm is 1.6 longer than the right one.

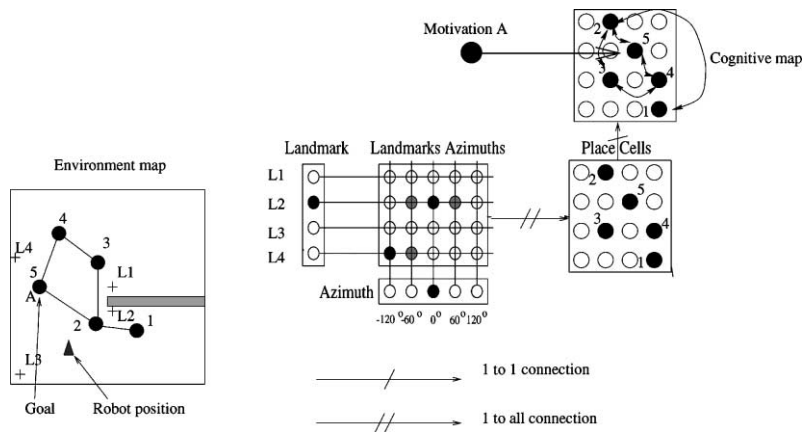


Fig. 2. Global sketch of the architecture. An animat in a simulated environment (on the left) and the corresponding landmark/azimuth fusion, place cell map and cognitive map. The cognitive map is projected on the environment in order to see where the place cells have been learned, and that are the links between them.

of its neighbors times the connection weight. So the activity of a neuron at a “distance” of N links from the activate goal is W^N , if all weights have the same value W . Taking the maximum enables to select one of the goals and suppresses local minima problems [24].

The kind of map built (Fig. 2) this way is fixed in the sense that the connection between the neurons may not change. Hence, for instance, when the environment changes (a door opens, somebody is walking, etc.) the map may not be used anymore. So it is necessary to be able to modify the links and/or to create/delete neurons when appropriate. For the time being, we have only investigated the first possibility. So, we have defined a hebbian learning rule reinforcing the links between neurons often activated successively and decreasing them if the link is not used. Note that this rule is not a “one shot learning”. The links have to be used several times before significantly increasing or decreasing. So when a change occurs in the environment, it may take some time before the new map is updated.

Let $W_{i,j}$ be the weight associated with the fact that from place P_i it is possible to reach directly place P_j , its learning rule is the following:

$$\frac{dW_{i,j}}{dt} = -\lambda W_{i,j} + \left(1 + \frac{dR}{dt}\right) (1 - W_{i,j}) \overline{\text{Pot}_{P_i}} \text{Pot}_{P_j} \quad (3)$$

Pot_{P_j} is the activity of place cell P_j . It is 0 if the animat is not at location P_j and non-zero and positive

otherwise (the exact value depends on the recognition of landmarks and may be also modulated by some motivation). Increase of weights $W_{i,j}$ occurs when Pot_{P_i} and Pot_{P_j} are non-zero. Since the animat cannot be at both locations simultaneously, we have to maintain the activity of place cell P_i until P_j is reached. This is performed by a time integration of the Pot_{P_i} values represented in the equation by $\overline{\text{Pot}_{P_i}}$. $\overline{\text{Pot}_{P_i}}$ decreases with time and can be used as a raw measure of the distance between i and j . λ is a very low positive value. It allows to forget unused links ($\text{Pot}_{P_i} = 0$ and $\text{Pot}_{P_j} = 0$). The term dR/dt corresponds to the variation of an external reinforcement signal (negative or positive) that appears when the animat enters or leaves a “difficult” or “dangerous” area. They model places where the ground may be of a different texture or containing very small obstacles.

4. Results

We are studying the way the animat behaves as the size of the left arm increases. In the beginning of each experiment, we choose the ratio “left arm/right arm”. The animat starts from its nest, which is located at the bottom of the vertical bar of the T-maze (Fig. 1). There are two phases in the experiment:

- the “discovery” phase, in which the animats has to discover the three sources; it then can choose either the left or the right food source when it is hungry;

- after its paths are stabilized (after 50 “meals”), we record the animat choices for the next 100 “meals”.

Each experiment is run during 50,000 steps, and has been tested 30 times. We give below the average choice rate for the left food source as the left arm length increases.

Whatever the size of the left arm, the animat has to go in it because it contains the only water source. However, it may eat either in the left or in the right arm. The basic expected behavior is to go to eat in the nearest food source when the eating motivation is triggered. As the animat is more often in its nest, it will go to the food source which is nearer to the nest. So as the left arm size increases, the right food source will be preferred.

A “smarter” behavior would be to go to eat near the place where the water source is. Indeed, this could save energy because if the animat gets thirsty while traveling to the food source, it would be also on the way to the water source. Note that there is no rule in our system able to foresee such a behavior.

On a symmetrical maze, we observed that, after some time, the animat almost always chooses to eat in the left arm. This is due to the *coupling* after some time of the eating and drinking motivations. This coupling occurs as soon as the animat eats just after (or before) drinking, so that the two levels are approximately the same.

The effect of this coupling is reported in a first experiment. We compare the animat behavior when (de-)activating the coupling between food and water needs. This is achieved by “forcing” it to drink each time it eats, so that it is never thirsty when it is hungry again.

We make the left arm grow from 1.0 to 1.8 times the initial size. The results are graphically represented in Fig. 3.

As can be seen on the results, coupling leads to an increased preference for the left food source. However, we chose to only reduce, and not fully suppress, this coupling since the need for water still happens from time to time. If the need for water was completely suppressed, the left arm selection rate would decrease from 50 to 0% as the size of the left arm increases. So this kind of smart behavior may be achieved by the sole coupling between internal needs and the environment.

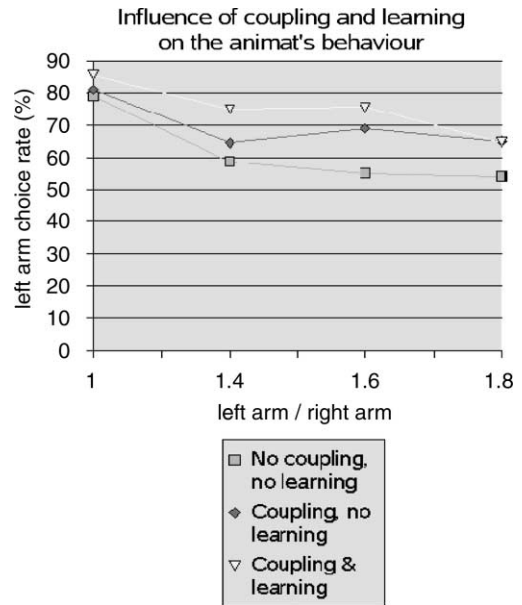


Fig. 3. Influence of coupling and learning on the left arm selection rate. The top curve (triangles) is with coupling and learning. The intermediate curve (diamonds) is with coupling and without learning, and the bottom curve (squares) is without coupling and learning. Enabling coupling leads to choose more often the left arm. Allowing learning still increases the choice of the left arm.

Now, we activate the learning rule on the links of the cognitive map. This rule increases the weight between two Place Cells activated successively. So, the more used the path, the higher the link. Conversely, a path less often used will see its link value decrease. As a consequence, when a motivation is activated, the source, which is nearer in terms of number of links from the actual position, is not necessarily the one providing the highest activity. So the animat may choose to go to a place which is farther away in terms of Euclidean distance, but closer in terms of cognitive map representation. So there is a remapping of the perception of the environment depending on the experience of the animat. In particular, paths often used will be reinforced, so that they will be perceived as “shorter”.

So, as shown in the following experiments (Fig. 3), the consequence of the learning rule is to keep choosing the left arm for eating, thus still following the “smart” behavior.

5. Conclusion and perspectives

The planning mechanism alone is not enough to account for smart behaviors. Their emergence, in our experiment, is first due to the coupling between the different motivational variables. This emphasizes the need for taking into account the internal dynamics of the animat and its coupling with the environment. Adding a learning rule on the planning map contributes to reinforce this behavior.

The learning rule we have described may be used in the more general context of a changing environment. Indeed, when the environment changes, new place cells are learned corresponding to the new landmark configuration. Their attraction basin extends over the previously created ones. In the extreme case, a place cell learned in the old environment may not fire at all in the new one. Since the place cell is not activated, it is as if it does not exist. So it will neither be linked to other place cells anymore, nor will the links be reinforced. However, if the old environment is presented again, the old place cells fire again, so that their links are valid again. So several different “layers” of cognitive maps can appear in the same physical neural network structure [33]. They may be linked together through some place cells, and may be activated successively when the environment they are coding is presented again. However, recent neurobiological findings rather support the evidence of maps stored for a short time at the hippocampal level which are transferred in the cortex areas for long time storage [9,27]. When the animal is brought back in the original environment, the same cognitive map appears, as if it were “uploaded” from the cortical areas.

The animat has to keep trying for some time before building a new efficient planning map. So, even if it may be a good solution to use a very low passive decay (λ parameter) to store several different maps (memory effect), this also slows down the process of finding new pathways, when one may not be used anymore. Indeed, the hebbian learning rule we have chosen needs some time before significantly changing the weights. Hence, in order to react faster to a change in the environment, it would be necessary to introduce an active decay mechanism decreasing unused links.

This navigation model does not take into account a path integration mechanism. This mechanism allows an animal to find the direct path to a particular location even if this movement has not been experienced before (shortcut). Our mechanism only allows to reinforce known paths and is not able to try a direct way to the goal if it is not an already existing path.

It is found that a deadlock situation occurs when a place field of a neuron further from the goal overlaps the trajectory to the goal (Fig. 4). Thus when the animat goes into this place field, it recognizes a location farther away from the goal (though topologically nearer on the planning map) and goes back from where it came. When it leaves the place field, it goes back towards the goal again. It follows the same trajectory as previously, and so eventually ends in the same overlapping place field. We are now back to the previous situation where the animat recognizes a location which is (for him) farther away from the goal. So, we have now a deadlock situation. Creating a new place cell in this situation does not resolve the problem since it would only be linked with place cell 1 (see Fig. 4). Thus, for the moment when a deadlock

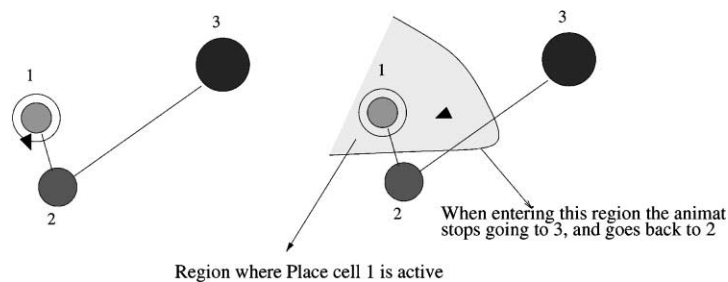


Fig. 4. Deadlock behavior. On the left, the animat is at position 1 heading for position 2 which corresponds to the most active neighbor. Once at 2, the animat goes to 3. However, while traveling to 3, the animat goes through the area where place cell 1 fires (figure on the left). Hence, the animat goes back to position 1 and begins to circle in a deadlock situation.

situation is detected, the animat switches to a random walk for some time before resuming the planning behavior. It should be noted that this kind of deadlock behavior mainly occurs because some links are missing: i.e. the cognitive map is not explored completely. For instance, a link between neuron 1 and 3 in Fig. 4 would lead to a correct planning behavior. Path integration could also be used for going directly from neuron 1 to neuron 3.

Finally, we have only considered in this work Place Cells in the classical approach: coding for a particular location in the environment. In the robotics experiments we have also carried out, we use these neurons for coding *transitions* between locations. This enables to associate such a neuron with the movement performed and given by a path integration mechanism [4]. So, path integration and transition coding are the two next improvements to the present simulation platform.

References

- [1] P. Andry, P. Gaussier, S. Moga, A. Revel, J. Nadel, Imitation: Learning and communication, in: Proceedings of the Sixth International Conference on the Simulation of Adaptive Behaviour (SAB 2000), Paris, September 2000.
- [2] M.A. Arbib, I. Lieblich, Motivational learning of spatial behavior, in: J. Metzler (Ed.), *Systems Neuroscience*, Academic Press, New York, 1977, pp. 221–239.
- [3] R.C. Arkin, Motor schema-based mobile robot navigation, *International Journal of Robotics Research* 8 (4) (1990) 92–112.
- [4] V. Babeau, P. Gaussier, C. Joulain, A. Revel, J.P. Banquet, Merging visual place recognition and path integration for “cognitive” map learning, in: Proceedings of the Sixth International Conference on the Simulation of Adaptive Behaviour (SAB 2000), Paris, September 2000.
- [5] I.A. Bachelder, A.M. Waxman, Mobile robot visual mapping and localization: A view-based neurocomputational architecture that emulates hippocampal place learning, *Neural Networks* 7 (6–7) (1994) 1083–1099.
- [6] I.A. Bachelder, A.M. Waxman, A view-based neurocomputational system for relational map-making and navigation in visual environments, *Robotics and Autonomous System* 16 (2–4) (1995) 267–289.
- [7] J.P. Banquet, P. Gaussier, J.C. Dreher, C. Joulain, A. Revel, Space–time, order and hierarchy in fronto–hippocampal system: A neural basis of personality, in: *Cognitive Science Perspectives on Personality and Emotion*, Elsevier, Amsterdam, 1997.
- [8] R.E. Bellman, On a routing problem, *Quarterly of Applied Mathematics* 16 (1958) 87–90.
- [9] B. Bontempi, C. Laurent-Demir, C. Destrade, R. Jaffard, Time-dependent reorganization of brain circuitry underlying long-term memory storage, *Nature* 400 (1999) 671–675.
- [10] G. Bugmann, A connectionist approach to spatial memory and planning, in: L.J. Landau, J.G. Taylor (Eds.), *Basic Concepts in Neural Networks: A Survey, Perspectives in Neural Networks*, Springer, Berlin, 1997, pp. 109–146.
- [11] G. Bugmann, J.G. Taylor, M.J. Denham, Route finding by neural nets, in: J.G. Taylor (Ed.), *Neural Networks*, Alfred Waller Ltd., Henley-on-Thames, 1995, pp. 217–230.
- [12] N. Burgess, M. Recce, J. O’Keefe, A model of hippocampal function, *Neural Networks* 7 (6–7) (1994) 1065–1081.
- [13] R. Chatila, Deliberation and reactivity in autonomous mobile robots, *Robotics and Autonomous System* 16 (2–4) (1995) 197–211.
- [14] T.S. Collett, J. Zeil, The selection and use of landmarks by insects, in: *Orientation and Communication in Arthropods*, Birkhäuser, Basel, 1997, pp. 41–65.
- [15] C.I. Connolly, J.B. Burns, R. Weiss, Path planning using Laplace’s equation, in: Proceedings of the International Conference on Robotics and Automation, Cincinnati, OH, 1990, pp. 2102–2106.
- [16] M.J. Denham, J. Boitano, A model of the interaction between prefrontal cortex, septum and the hippocampal system in the learning and recall of goal-directed sensory-motor behaviours, Technical Report NRG-96-01, School of Computing, University of Plymouth, 1996.
- [17] J.Y. Donnart, J.A. Meyer, Learning reactive and planning rules in a motivationally autonomous animat, *IEEE Transactions on Systems, Man and Cybernetics Part B* 26 (3) (1996) 381–395.
- [18] M.O. Franz, B. Schölkopf, H.H. Bülthoff, Homing by parameterized scene matching, in: Proceedings of the Fourth European Conference on Artificial Life, MIT Press, Cambridge, MA, 1997, pp. 236–245.
- [19] M.O. Franz, B. Schölkopf, H.A. Mallot, H.H. Bülthoff, Learning view graphs for robot navigation, *Autonomous Robots* 5 (1998) 111–125.
- [20] M.O. Franz, B. Schölkopf, H.A. Mallot, H.H. Bülthoff, Where did I take that snapshot? Scene-based homing by image matching, *Biological Cybernetics* 79 (1998) 191–202.
- [21] O. Gallet, P. Gaussier, J.P. Cocquerez, A parallel singularity amplifier model for focusing visual attention, in: B. Dresp, C. Bonnet (Eds.), *European Conference on Visual Perception (ECVP’96)*, Congrès Pasteur, Strasbourg, September 1996, p. 74.
- [22] P. Gaussier, C. Joulain, J.P. Banquet, S. Leprêtre, A. Revel, The visual homing problem: An example of robotics/biology cross fertilization, *Robotics and Autonomous Systems* 30 (2000) 155–180.
- [23] P. Gaussier, C. Joulain, S. Zrehen, J.P. Banquet, A. Revel, Visual navigation in an open environment without map, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS’97), Grenoble, France, September 1997, pp. 545–550.
- [24] P. Gaussier, S. Leprêtre, M. Quoy, A. Revel, C. Joulain, J.P. Banquet, Experiments and models about cognitive map learning for motivated navigation, in: *Interdisciplinary*

- Approaches to Robot Learning, Robotics and Intelligent Systems Series, Vol. 24, World Scientific, Singapore, 2000, pp. 53–94. ISBN 981-02-4320-0.
- [25] P. Gaussier, S. Zrehen, Navigating with an animal brain: A neural network for landmark identification and navigation, in: Proceedings of the Intelligent Vehicles, Paris, 1994, pp. 399–404.
- [26] S. Leprêtre, P. Gaussier, J.P. Cocquerez, From navigation to active object recognition, in: Proceedings of the Sixth International Conference on the Simulation of Adaptive Behaviour (SAB 2000), Paris, France, September 2000.
- [27] N. Ludvig, Place cells can flexibly terminate and develop their spatial firing: A new theory for their function, *Physiology and Behavior* 67 (1) (1999) 57–67.
- [28] M.J. Matorić, A distributed model for mobile robot environment learning and navigation, Technical Report TR 1228, MIT AI Laboratory, Cambridge, MA, 1990.
- [29] S. Moga, P. Gaussier, A neuronal structure for learning by imitation, in: D. Floreano, J.D. Nicoud, F. Mondada (Eds.), *Advances in Artificial Life. 5th European Conference ECAL'99* Lausanne, Switzerland, September 13–17, 1999, Lecture Notes in Artificial Intelligence, Vol. 1674, Springer, Berlin, 1999, pp. 314–318.
- [30] U. Nehmzow, T. Smithers, Mapbuilding using self-organising networks, in: J.A. Meyer, S. Wilson (Eds.), *From Animals to Animats: SAB'91*, MIT Press, Cambridge, MA, 1991.
- [31] J. O'Keefe, N. Nadel, *The Hippocampus as a Cognitive Map*, Clarendon Press, Oxford, 1978.
- [32] M. Quoy, S. Moga, P. Gaussier, A. Revel, Parallelization of neural networks using PVM, in: J. Dongarra, P. Kacsuk, N. Podhorszki (Eds.), *Recent Advances in Parallel Virtual Machine and Message Passing Interface*, Lecture Notes in Computer Science, Vol. 1908, Springer, Berlin, 2000, pp. 289–296.
- [33] A. Redish, D. Touretzky, Separating hippocampal maps, in: K. Jeffery, N. Burgess, J. O'Keefe (Eds.), *The Hippocampus and Parietal Foundations of Spatial Cognition*, Monte Verita, Ticino, Switzerland, Oxford University Press, Oxford, 1999.
- [34] A. Revel, P. Gaussier, J.P. Banquet, Taking inspiration from the hippocampus can help solving robotics problems, in: *European Symposium on Artificial Neural Networks*, Bruges, Belgium, IEEE Press, New York, 1999.
- [35] A. Samsonovich, B. McNaughton, Path integration and cognitive mapping in a continuous attractor neural network model, *Journal of Neuroscience* 17 (1997) 5900–5920.
- [36] N.A. Schmajuk, H.T. Blair, Place learning and the dynamics of spatial navigation: A neural network approach, *Adaptive Behavior* 1 (1992) 353–385.
- [37] N.A. Schmajuk, A.D. Thieme, Purposeful behavior and cognitive mapping: A neural network model, *Biological Cybernetics* 67 (1992) 165–174.
- [38] B. Schölkopf, H.A. Mallot, View-based cognitive mapping and path-finding, *Adaptive Behavior* 3 (1995) 311–348.
- [39] E.C. Tolman, Cognitive maps in rats and men, *The Psychological Review* 55 (4) (1948).
- [40] O. Trullier, S.I. Wiener, A. Berthoz, J.A. Meyer, Biologically based artificial navigation systems: Review and prospects, *Progress in Neurobiology* 51 (1997) 483–544.
- [41] U.R. Zimmer, Robust world-modelling and navigation in a real world, *Neurocomputing* 13 (1996) 247–260.



Mathias Quoy received a computer science engineering degree in Toulouse (France) in 1990, and a PhD in computer science in 1994. He is now Associate Professor at the Cergy-Pontoise University where he teaches computer science. He is member of the neurocybernetics group of the ETIS lab. His main research interests are the development of neural architectures for the control of mobile robots, and the study of dynamical neural networks.



Philippe Laroque got a PhD in computer science in 1987. He is now Associate Professor at the Cergy-Pontoise University where he teaches computer science. He is member of the neurocybernetics group of the ETIS lab. His main research interest is the development of learning techniques using neural architectures.



Philippe Gaussier received the MS degree in electronic from Aix-Marseille University in 1989. In 1992, he received a PhD degree in computer science from the University of Paris XI (Orsay) for a work on the modeling and simulation of a visual system inspired by mammals vision. From 1992 to 1994, he conducted research in Neural Network (NN) applications and in control of autonomous mobile robots at the Swiss Federal Institute of Technology. He has edited a special issue of *Robotics and Autonomous System Journal* on “moving the frontier between robotics and biology”. He is now Professor at the Cergy-Pontoise University in France and leads the neurocybernetic team of the Image and Signal processing Lab (ETIS). His research interests are focused on one hand on the modeling of the cognitive mechanisms involved in visual perception, motivated navigation and action selection, and on the other hand on the study of the dynamical interactions between individuals (imitation capabilities, collective intelligence, social interactions, ...).