HAL
open science

Path Integration Working Memory for Multi Task Dead Reckoning and Visual Navigation

Cyril Hasson, Philippe Gaussier

To cite this version:

Cyril Hasson, Philippe Gaussier. Path Integration Working Memory for Multi Task Dead Reckoning and Visual Navigation. Simulation of Adaptive Behavior'10, Aug 2010, Paris, France. pp.380-389. hal-00538397

HAL Id: hal-00538397

https://hal.science/hal-00538397v1

Submitted on 22 Nov 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Path Integration Working Memory for multi tasks dead reckoning and Visual Navigation

Cyril Hasson, Philippe Gaussier

Université de Cergy-Pontoise, CNRS, ENSEA
ETIS laboratory UMR 8051
F-95000 Cergy Cedex, France

Abstract. Biologically inspired models for navigation use mechanisms like path integration or sensori-motor learning. This paper describes the use of a proprioceptive working memory to give path integration the potential to store several goals. Then we coupled the path integration working memory to place cell sensori-motor learning to test the potential autonomy this gives to the robot. This navigation architecture intends to combine the benefits of both strategies in order to overcome their drawbacks. The robot use a low level motivational system. Experimental evaluation is done with a robot in a real environment.

1 Introduction

Researchs in the field of navigation robotics have used biologicaly inspired mechanisms like path integration based on odometric information [1–3] (return vector computing) or sensori-motor learning based on place cell recognition [4–6]. These navigation strategies are very good solutions to homing problems. However, following the animat approach [7], to ensure its survival, a robot must maintain a set of artificial physiological variables inside safe levels. Thus, the robot must look for different resources to fullfill its various needs. Path integration doesn't need learning, but, by itself, it is not able to store several goals. And errors, coming from measure imprecision, cumulate to the point where it is too inaccurate to allow the robot to find its goal. Sensori-motor learning is robust but needs learning (generaly man supervised). However, study about insects navigation [8, 9] have shown their ability to manage several homing vectors allowing them to go back to a secondary goal when their first one is not available (indicating a memory).

In this paper, we describe a proprioceptive working memory giving path integration abilities the potential to allow robots to display the same kind of behaviors. The robot control architecture use a low level motivational, or *drive* system. It reacts to the physiological state and computes the different drives levels. From the principles presented in [1], we have designed a proprioceptive navigation strategy using this working memory to store several goals on several path integration neural fields. The robot uses hebbian learning to associate each goal to the corresponding drive. Furthermore we then coupled place cell sensori-motor learning to

the path integration working memory and the drive system. Place-drive-action associations are used to autonomously build a visual attraction bassin around each goal and allowing to bypass the limitations of path integration when the robot is lost or has been kidnapped.
Section 2 describres the proprioceptive navigation architecture we used. The visual place cell architecture is described in section 3. Section 4 shows experimental results with the robot. And section 5 contains the conclusions. Figure 1 shows the robot and its environment.

Fig. 1. The robot in its environment (equiped with a color detector placed under it). Colored squares on the ground are simulated resources.

2 Proprioceptive navigation

Path integration is the ability to use proprioceptive information about the movements being done in order to determine the direct movement to any given interesting point of the robot trajectory in the environment. Principles of path integration using dynamical neural fields are described in [1]. Figure 2 is an illustrated example of this computation. This strategy is said to be autonomous because the robot can learn and use it without any supervision. Using Dynamical neural field to represent information in the path integration process allows to use directly the output (the return vector) as the control signal for the robot rotational speed. In order to use path integration to build navigation abilities able to succeed to classical multigoal survival tasks (as presented in introduction), the robot has to be able to come back to several interesting places of its environment (vital resources locations). Thus, instead of only one integration field, the robot must dispose of several integration fields. But because the number of path integration field has to be limited (defining a more realistic kind of working memory), the robot cannot simply recruit another integration field each times it detects a resource. The number of parallel integration fields (nb_{goals}) is a representation of the system working memory span i.e. the number of elements that can be maintained

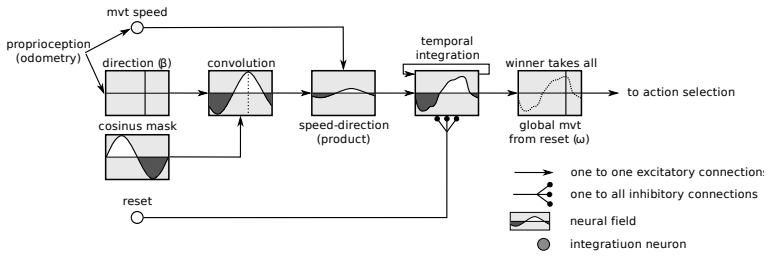

Fig. 2. Path integration : speed is coded as the activity of one neuron and orientation as the most active neuron of a neural field. At every time step, the integrator takes as input the activity of the orientation neural fiels (convoluted by a cosine shape) multiplied by the activity of the speed neuron. This input represents the orientation and distance traveled since the last time step. Summing this input with its own activity, the integration neural field computes the return vector. When integration

in working memory). Activity of each neuron of these parallel fields at time t is $P_{ij}(t)$:

$$P_{ij}(t) = \sum_{t_{rj}}^{t} (S(t) \ . \ cos(\frac{d_w(t) - i}{n})) \ . \ (1 - r_j(t))$$

n is the size of the neural fields, $i \in [1 : \mathrm{n}]$, j is the number of path integration fields ($j \in [1 : nb_{goals}]$), t_{rj} is t at the last field j reset, $S(t)$ is the activity of the speed coding neuron at time t, $d_w(t)$ is the active direction neuron position in the field at time t and $r_j(t)$ is the reset signal for the field j at time t (1 during reset, 0 otherwise). A cosine function has been used, but can be replaced by any bell curve activity i.e. (a gaussian).
When the robot finds a new resource, it must be able to recruit a new integration field. And when it is motivated by its simulated physiological needs, the robot must be able to select among the different integration fields the best to lead it to the desired resource. We will describe how this can be done using a modified version of the simulated neural networks used for simple path integration.

New goal / known goal discrimination :
To exploit optimally this multiple field path integration architecture, it is important that the robot discriminates new resource locations from known ones. Every time a resource is detected, one of the integration fields must be reset (all neurons in the field have a null activity). Reseting a new integration field when it is a new resource (recruitment) and reseting the corresponding integration field when it is a known resource (recognition). To discriminate new from known goals, we use the distance coding property of the integration fields (neural field maximum potential). A group of neurons coding for goals proximity (size $= nb_{goals}$) receives activation from a contant input and each of the neurons is inhibitied by its corresponding positive activity in the integration field. As the robot gets closer to a known goal, activity of the corresponding goals proximity neuron gets higher. if we use neurons with a non-linear

transfert function (here a simple just below 1 thresold), a goal proximity neuron will only be active when a known resource is near. This activity could be seen as a goal prediction or expectation. Activity of each goal prediction neuron at time t is Goal Prediction$_j(t)$:

$$Goal\ Prediction_j(t) = \begin{cases} 1 & if\ (1\ -\ w_{j'j}\ .\ \sum_{i=1}^{i=n} |(P_{ij}(t))|)\ >\ T \\ 0 & otherwise \end{cases}$$

$j \in [1 : nb_{goals}]$, $j' \in [1 : nb_{goals}]$, $w_{j'j}$ is the weight of the path integration field$_{j'}$−goal proximity$_j$ (small negative value), 1 is the constant input, n is the size of the neural integration fields and T is a definite threshold of the form (1 - ϵ).
Figure 3 shows how this goal prediction is used to discriminate new from known goals. Resource detection both activates the new goal and the know goal neurons but goals predictions neurons inhibit the new goal neuron and the new goal neuron inhibits the know goal neuron. Thus when a resource is detected, if no goal prediction is made, this resource is considered as a new goal (a known goal otherwise).

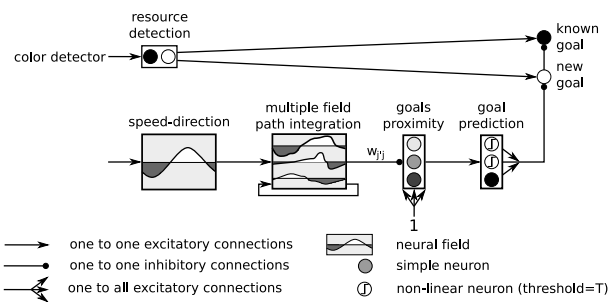

Fig. 3. New goal/known goal discrimination. As the robot gets closer to a known goal, the corresponding goal proximity neuron activity gets closer to 1. As shown here, above a definite threshold (T = 1 - ϵ), a goal prediction is made and resource detection will then be considered as detection of a known goal rather than of a new goal.

Integration field recrutment and goal recognition :
When a new goal is detected, a new integration field must be recruited. This is done by resetting one of the integration fields. Figure 4 (upper part) shows how fields to be recruited are selected.The main idea is to take an unused field or at least the field associated to the least used goal. The "most used goals" group of neurons (size = nb_{goals}) receives one to one connections from the recruitment reset group of neurons (same size) and has recurrent one to one connections with a weight slightly under 1. Each time a new integration field is recruited, the corresponding "most used goals" neuron receives activation and the reccurent connections act as a decay function. Thus, the neuron of the "most used goal" group corresponding to a newly recruited field will be more active than the one of an old goal. The "less used goal" group of neurons is a winner takes all groupe of neurons that receives a constant activation input (one to all connections) and is inhibited by the "most used goals" group of neurons (one to one connections). Its single active neuron thus corresponds to the integration field to inhibit when a new goal is detected. The recruitment reset group simply makes the product of the

"less used goal" activities (one to one connections) and the new goal detection neuron activity (one to all connections). Only one neuron of the recruitment reset group can be active at a given time, and only when a new goal is detected. Each of its neurons inhibits an entire field of the multiple path integration fields group. When a known goal

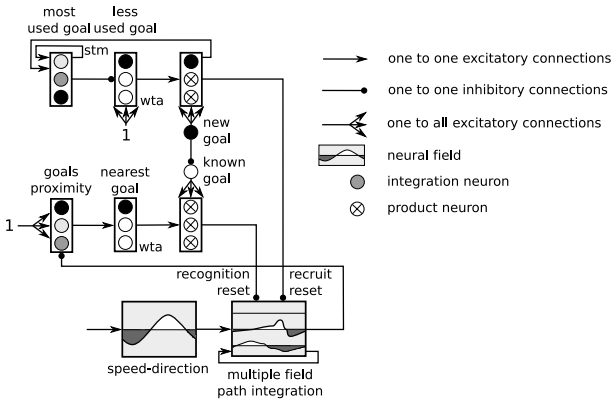

Fig. 4. Integration field recruitment and goal recognition. As shown in this example, when a new goal is detected, the integration field corresponding to the less used goal is reset (recruitment). When a know goal is recognized, the integration field corresponding to the nearest known goal is reset (recognition)

is detected, the corresponding integration field should have no activity. However, because the resources are represented by square surfaces the robot might detect a known resource from a position different from the reset point and a little activity might still be found on the corresponding integration field. Furthermore, a residual activity on the field might be caused by integrations errors due to discretisation or even by the sliding of the robot wheels on the floor. To avoid the cumulative effect of these errors, when a known goal is recognized, the corresponding integration field is also reset inducing a recalibration effect similar to [10]. Figure 4 (lower part) shows how fields to be reset because of goal recognition are selected. The nearest goal group is a winner takes all group of neurons that receives one to one connections from the goals proximity group. The only active neuron corresponds to the closest goal. The recognition reset group simply makes the product of the nearest goal group activities (one to one connections) and of the known goal detection neuron activity (one to all connections). Only one neuron of the recognition reset group can be active at a given time. And only when a known goal is detected. Each of its neurons inhibits an entire field of the multiple path integration fields group. The recognition reset group projects activation one to one connections to the most used goals group so that a goal is considered as used when it is recruited as well as when it is recognized.

Goals competition and integration field selection :
Once several integrations field have been recruited, it is important to be able to select the right one to reach the desired resource. The architecture can only work if it is able to learn the association between the goals (and their corresponding integration fields)

and the drive they satisfy. Furthermore, one resource can be present in several different locations of the environment. It is then necessary to select one of these locations (e.g. according to their distances). Figure 5 shows the neural network used to achieve this. The goal-drive association group receives one to one inconditionnal connections from

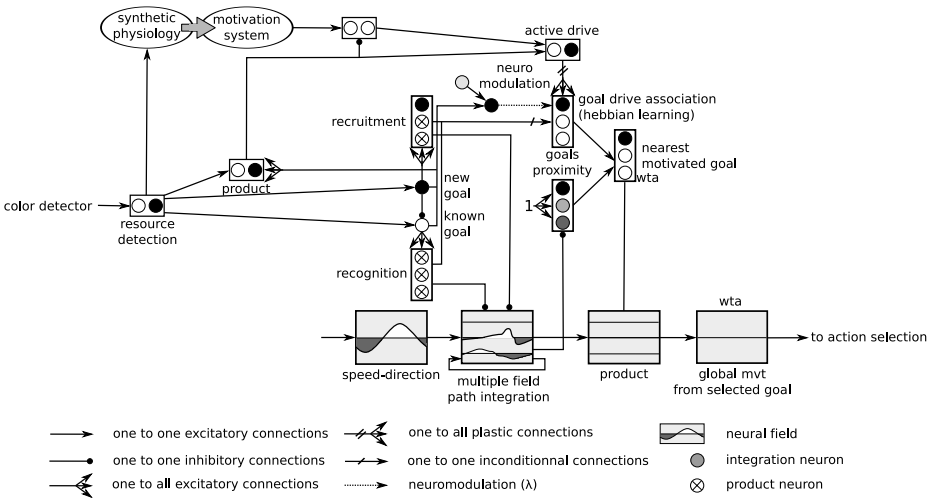

Fig. 5. Goals competition and integration field selection. The goal drive association group learns which goal satisfies which drive and the nearest motivated goal group select the nearest motivated goal. The corresponding integration field (and thus the corresponding action) is selected via a neuronal matricial product.

the recruit reset group and the recognition reset group and one to all plastic connections from the active drive group. Following its hebbian learning rule, weights of the plastic connections will adapt so that when a drive is active, the goal-drive association group will have activity on the neurons corresponding the goals that satisfy this drive. Every time a goal is detected, the corresponding goal-drive association is reinforced. Activity of each goal-drive neuron at time t is $GD_i(t)$:

$$GD_i(t) = \sum_{1}^{j} D_j(t) \ . \ w_{ji}(t)$$

Weights adaptation :

$$\Delta w_{ji}(t) = \lambda(t) \ . \ (recruitR_i(t) \ + \ recogR_i(t)) \ . \ D_j(t)$$

i is working memory span size (nb_{goals}), j is the number of drive, $D_j(t)$ is the active drive neuron j activity at time t, $w_{ji}(t)$ is the weight of the D_j - GD_i connexion at time t, $\lambda(t)$ is the learning rate at time t, $recruitR_i(t)$ and $recogR_i(t)$ are the recruitment reset and recognition reset signals for goal i at time t (reset of a neural field when a new goal and a know goal are detected). Neuromodulation of this hebbian learning group of neurons (λ) is high when a goal is detected allowing fast learning of the goal-drive association. It is low otherwise, letting slowly forget the drive associations of goals that

could satisfy the active drive but are not detected).
To take into account selection by the drive as well as selection by goal distance, the nearest motivated goal group of neurons receives activations one to one connections from the goal-drive association as well as from the goals distances group and sums its inputs. Using the winner takes all rule, its single active neuron corresponds to the closest goal which satisfy the active drive. Selection of the corresponding integration field is done by a matrix product between the multiple integration fields group and the nearest motivated group. This matrix product is done in two steps. First, a group of neurons the same size as the multiple integration fields group makes the product between the activity it receives from the multiple integration fields group (one to one connections) and from the nearest motivated goal group (one to a field connections i.e. horizontal projections). The field corresponding to the selected goal is the only one to sustain activity (the other fields have an activity which is product by 0). Finally, this almost empty neurons matrix is projected through vertical connections (vertical projection) to a group with a single field of neurons which can then be used just like in the simple path integration model to compute the direction of the return vector.

Path integration benefits and drawbacks :
Because the robot only needs to detect a resource once to be able to store its return vector, this navigation strategy doesn't need learning. It gives an autonomous ability to the robot. When it is not motivated, the robot explores randomly its environment. If a drive is high enough, it will then be able to reach the resources locations it has discovered. However, path integration has a major drawback. It is not precise over long periodes of time. Cumulative errors come from the direction discretization and from the slidings of the robot wheels on the floor. Studies of path integration on different animals [11] have shown that these cumulative errors are structural limitations.

3 Visual navigation

The visual system is able to learn to characterize (and thus recognize) different "places" of the environment. Inspired by visual navigation models issued from neurobiology [12], the visual system, a simulated neural network, learns place cells. Each place cell codes information about a constellation of local views (visual cues) and their azimuths from of a specific place in that environment [4, 13]. Activities of the different place cells depend on the recognition levels of these visual cues and of their locations. A place cell will then be more and more active as the robot gets closer to its learning location. The area where a given place cell is the more active is called its place field. When the maximum recognition level of place cells is below a given threshold T, another place is learned. The higher is T, the more place cells are learned in a given environment. An associative learning group of neurons allows sensorimotor learning (the place-drive-action group on figure 6). Place-drive neurons are associated with the return vector of the corresponding goals to autonomously build a visual attraction bassin around each goal. Figure 6 shows how this navigation strategy works and how it allows different responses according to the active drive. The main advantage of this navigation strategy over path integration is that it is not sensitive to cumulative errors. Learning precision can be maintained over long periodes of time. However, this strategy is hard to learn autonomously. The perceptual context is defined in a more stable way than for path integration, but, because the robot cannot see the resource on the floor, information about goal direction is not available. Learning is thus traditionally supervised.

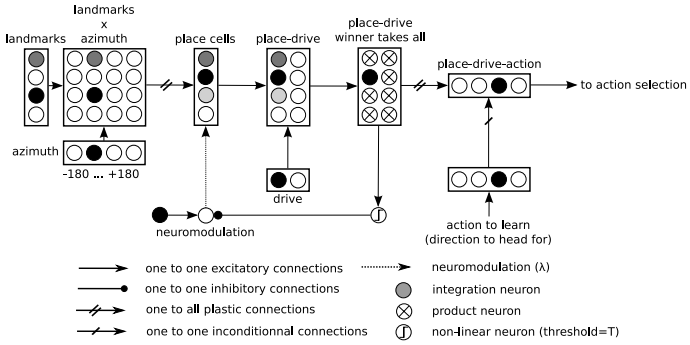

Fig. 6. Sensorimotor visual navigation : a visual place cell is constructed from recognition of a specific landmarks-azimuths pattern and an action is associated with this place cell. The action to learn is usually given through supervised learning.

4 Robotic Experiments

The main goal of this proprio-visual navigation architecture is to take advantage of both strategies in order for the robot to be more autonomous both in terms of learning and robustness. The task is a multiple resource problem : the robot needs two different resources (water and food) and each resource is present in two different places of the environment (see figure 1). In the first experiment, we only used the proprioceptive strategy to learn the task. Figure 7 shows the robot trajectories after learning is made (the resource are discovered by random navigation).

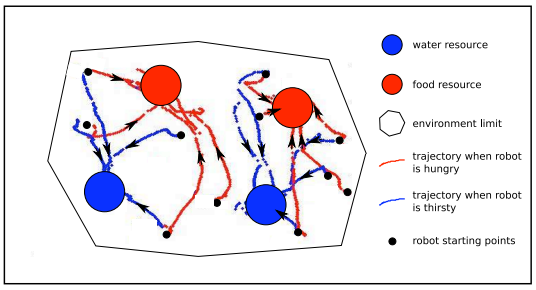

Fig. 7. Proprioceptive navigation trajectories : when motivated, the robot heads for the closest corresponding resource.

In the second experiment, we only used the visual strategy. But lacking crucial path integration information (goal direction), this strategy is not able to learn autonomously and the robot is driven by the noise in its neurons activity. Figure 8 shows the robot trajectories when using stand alone unsupervised visual navigation.
In the third experiment, both strategies were coupled in order for the visual strategy to use information coming from the proprioceptive strategy. During latent place-action

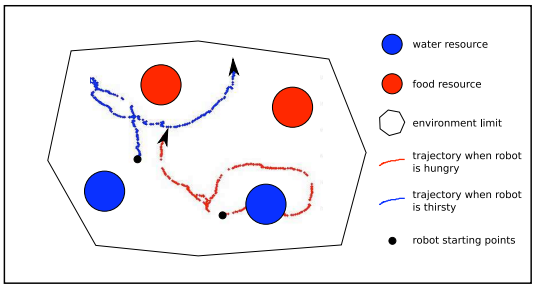

Fig. 8. Pace can be recognized but are not linked to actions. Without learning, the robot navigates randomly.

learning, the visual strategy associates place cell recognition to the return vector computed by the proprioceptive strategy. Figure 9 shows the robot trajectories using the visual strategy coupled to path integration (10 minutes of proprioceptive navigation).

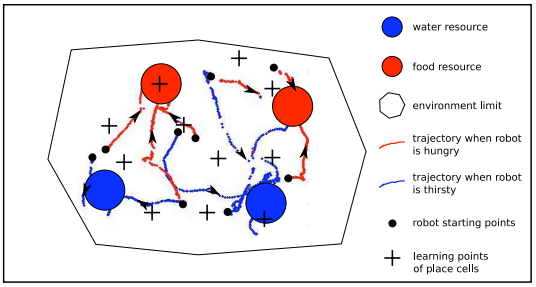

Fig. 9. Visual navigation trajectories : after 10 minutes of proprioceptive navigation, visual navigation has learned enough to produce a converging behavior. When motivated, the robot heads for the corresponding resource.

5 Conclusion and perspective

Complementary aspects of a motor working memory based on neural field associated to visual place recognition give a robot autonomous abilities for navigation. While proprioceptive navigation using a working memory is easy to learn, it is not robust and unless the robot constantly navigate beetween short distance goals, it will inevitably become less and less precise until it is not usable anymore. The experimental results show that the visual strategy needs proprioceptive information for an autonomous learning since the robot has no visual information about the places of the environment to look for. Visual strategy has the strong advantage of being robust over time allowing the recalibration of the path integration fields. The coupling of these two strategies allows to boostrap learning using the proprioceptive strategy and then to use the proprioceptive strategy output (an action) as input for the visual strategy sensori-motor learning.

However, further developments of this navigation architecture should be focused on the need to design a mechanism to select which strategy has to be used. This selection mechanism could rely on very different but nonetheless equally important parameters. The strategy to use could be selected according to its propension to satisfy the underlying drive (or motivation). In other words, a frustration mechanism based on prediction of the drive satisfaction could be very efficient to regulate autonomously the strategy to use and thus the robot behavior. Furthermore, each strategy being based on distinct information sources, strategy selection could rely on perceptive context.

References

1. Gaussier, P., Banquet, J.P., Sargolini, F., Giovannangeli, C., Save, E., Pousset, B.: A model of grid cells involving extra hippocampal path integration, and the hippocampal loop. Journal of Integrative Neuroscience (2007)
2. Vickerstaff, R.J., Paolo, E.A.D.: Path integration working memory for proprioceptive and visual navigation. Journal of Experimental Biology (2005)
3. Mittelstadt, M., Mittelstadt, H.: Homing by path integration in a mammal. Naturwissenschaften (1980)
4. Gaussier, P., Zrehen, S.: Perac: A neural architecture to control artificial animals. Robotics and Autonomous System (1995)
5. Gaussier, P., Joulain, C., Banquet, J., Leprtre, S., Revel, A.: The visual homing problem: an example of robotics/biology cross fertilization. Robotics and autonomous system (2000)
6. Wiener, S., Berthoz, A., Zugaro, M.: Multisensory processing for the elaboration of place and head direction responses in the limbic system. Cognitive Brain Research (2002)
7. Donnart, J.Y., Meyer, J.A.: Learning reactive and planning rules in a motivationally autonomous animat. Systems Man and Cybernetics (1996)
8. Gallistel, C.: Symbolic processes in the brain: The case of insect navigation. In: An Invitation to Cognitive Science: Methods, models, and conceptual issues. The MIT Press (1998)
9. Collett, M., Collett, S., Srinivasan, M.: Insect navigation : Measuring travel distance across ground and through air. Current biology (2006)
10. Arleo, A., Gerstner, W.: Spatial cognition and neuro-mimetic navigation: a model of hippocampal place cell activity. Biological Cybernetics (2000)
11. Etienne, A., Maurer, R., Seguinot, V.: Path integration in mammals and its interaction with visual landmarks. Journal of Experimental Biology (1996)
12. O'Keefe, J., Nadel, L.: The Hippocampus as a Cognitive Map. Oxford University Press (1978)
13. Gaussier, P., Lepretre, S., Quoy, M., Revel, A., Joulain, C., Banquet, J.: Experimentsandmodels about cognitive map learning for motivated navigation. In: Interdisciplinary Approaches to Robot Learning. Robotics and Intelligent Systems Series World Scientific (2000)