

Modeling of Spatial Navigation Inspired by Rodent Hippocampus

MICHAL BUREŠ¹, MARCEL JIŘINA²

¹Gerstner Laboratory, Department of Cybernetics
Czech Technical University in Prague
Karlovo náměstí 13, 121 35, Prague 2
CZECH REPUBLIC
<http://gerstner.felk.cvut.cz/>

²Institute for Biomedical Engineering
Czech Technical University in Prague
Zikova 4, 166 36, Prague 6
CZECH REPUBLIC
<http://ubmi.cvut.cz/>

Abstract: - This paper solves spatial navigation task inspired by methods of living organisms, especially by hippocampus that is responsible for spatial navigation. We attempt to model the behavior of a rodent hippocampus by methods of artificial neural networks and reinforcement learning theory. A model of mutual relations among several parts of hippocampal formation is suggested. To investigate the behavior of our model a task similar to the well-known Morris water maze task is considered [8].

Key-Words: - Hippocampus, spatial navigation, reinforcement learning, place cells, neural networks, animat.

1 Introduction

A preliminary indication that hippocampus maintains spatial information was described in [10]. Abilities of rat hippocampus responsibilities for spatial memory and navigation (orientation) have been found out on the experiments [11], [8]. The discovered place cells [9] of the hippocampus are active (firing) when the rat occurs in the corresponding particular place of the environment. Further research revealed another features of the hippocampus, e.g. head direction cells [17], modulation of long-term potentiation of synapses by θ rhythm [12], independence of CA1 place fields on the goal position [16]. More experimental results are to be found in [4].

In recent years, many computer models of spatial navigation of animals were published. Also, artificial systems, called animats, were designed. One of the first complex models was a multi-layer feed-forward neural network published in [5]. Sophisticated and detailed model of rodent hippocampus was described in [14], simulating such phenomena as θ rhythm. A mobile robot [1] involves unsupervised growing neural network to build up a map of the environment. Recently, RatSLAM [7] presented a competitive attractor network and was successfully tested within a large indoor environment.

2 Model

2.1 Model concept

Our model consists of several blocks (fig. 1) represented by a single- or multi-layer artificial neural networks. Sensory inputs (SI) are used to acquire and process information from the environment and to recognize actual animat's position (PR). Position is coded by both head direction (HD) system and place cells (PC). The path integrator (PI) updates supposed location during move. The navigational map (NM) is developed during

several successful episodes (reward (Rw) is simulated by reinforcement learning) and then the animat is capable to find out a way to the target starting at an arbitrary position. Reading in the navigational map is a role of the locomotion control (LC) subsystem. It sends motor signals, which are received by the motor (Mo), and looped back to the PI and HD systems.

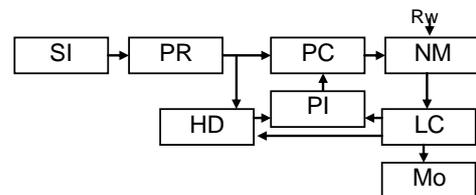


Fig. 1: Model structure: sensory inputs (SI), position recognition (PR), head direction (HD), place cells (PC), path integrator (PI), navigational map (NM), reward (Rw), locomotion control (LC), motor (Mo).

2.2 Allothetic sensory inputs

Sensory inputs (SI) consist of a highly processed signal. They detect position of environmental landmarks relative to the animat. This is the only allothetic (vision-based) information from the environment. Arena walls, target position or any obstacles in the environment aren't detected. We use several point landmarks [13], so that only position, but neither size nor orientation, are considered.

The sensory inputs are processed by a pre-wired one-layer RBF (radial basis function) neural network. Two types of neurons code relative position of the landmark. Activity of distance-sensitive neurons is given by distance of the landmark, whereas egocentric-bearing-sensitive neurons respond to the heading of the animat. For each landmark, a group of both egocentric-bearing-sensitive and distance-sensitive neurons exists. Similar information is supposed to come into the rodent's hippocampus [13].

Our animat detects all landmarks in the arena simultaneously, independently of their position or direction of the animat's heading. We can consider this as the animat stops moving and looks around. However, in a real world, it might be a time-consuming operation. We later describe mechanism based on the PI to reduce computational effort using the SI as much as possible.

2.3 Head-direction system

The head-direction (HD) system codes heading of the animat. It emulates head-direction cells [17] observed in the rodents' hippocampus. Activity of the HD neurons is tuned by heading of the animal. The HD neuron fires only when the animal is heading in neuron's preferred direction, regardless of the animal's location. Activity of HD cells forms an activity packet – an ensemble of neurons firing altogether [18].

We model HD system as a recurrent (Hopfield-like) neural network. Synaptic weights between HD neurons are pre-wired and aren't affected by learning process, but they're modulated by locomotion. When the animat is stopped, weights values between HD neurons are

$$w_{i,j}^{HD-0} = g(|\theta_i - \theta_j|), i \neq j; w_{i,i}^{HD-0} = 0, \quad (1)$$

where θ_i, θ_j are preferred angles of neurons i, j and $g(\cdot)$ is a distribution weight function. This function is requested to link two neurons the strongly the preferred angles are closer. For much different angles, we request the function to create inhibition or none link. We've chosen a simple Gaussian-like function; however, the form of this function is not critical for our model and it is also possible to use a linear function with saturations.

During locomotion, heading of the animat is changed. To shift coded direction left or right, respectively, we need to modulate synaptic weights by both heading and angular speed of the animat. Modulation is simulated by synaptic matrices \mathbf{W}^{HD-L} and \mathbf{W}^{HD-R} , respectively:

$$w_{i,j}^{HD-R} = \left. \frac{dg(\theta)}{d\theta} \right|_{\theta=|\theta_i-\theta_j|} \approx \frac{g(\theta - \frac{2\pi}{N_{HD}}) - g(\theta + \frac{2\pi}{N_{HD}})}{\frac{4\pi}{N_{HD}}}, \quad (2)$$

$$\mathbf{W}^{HD-L} = -\mathbf{W}^{HD-R}.$$

Synaptic matrix composing N_{HD} HD neurons consists of both static and modulating elements:

$$\begin{aligned} \mathbf{W}^{HD} &= \mathbf{W}^{HD-0} + \max(v_{HD}, 0) \cdot \mathbf{W}^{HD-R} + \max(-v_{HD}, 0) \cdot \mathbf{W}^{HD-L} \\ &= \mathbf{W}^{HD-0} + v_{HD} \cdot \mathbf{W}^{HD-R} \end{aligned} \quad (3)$$

Modulation coefficient v_{HD} is proportional to the angular speed of the animat. The HD system is capable to integrate the angular speed during locomotion. This information is supplied by vestibular (motor) signals. Hence, our animat doesn't need the sensory (allothetic) information to update the coded heading during move.

The output of the HD system is given by

$$\mathbf{A}^{HD}(t+1) = \sigma \left(K_{HD} \frac{\mathbf{W}^{HD} \cdot (\mathbf{A}^{HD}(t) + \mathbf{V}^{HD}(t))}{\frac{1}{N_{HD}} \sum \mathbf{W}^{HD} \cdot (\mathbf{A}^{HD}(t) + \mathbf{V}^{HD}(t))} \right), \quad (4)$$

where $\sigma(\cdot)$ is a modified sigmoid function, \mathbf{W}^{HD} is a synaptic matrix of the network, \mathbf{V}^{HD} represents influence of the SI (and is zero most of the time). Parameter K_{HD} has similar function as a gain in a loop-back system. The equation (4) provides a simple mechanism to control activity of the network: when the activity of the network is too high, the denominator of (4) becomes high and it effects attenuation in the network, and vice versa.

To get the most possible stable representation of the HD, we found a value of the parameter K_{HD} as a minimum of quadratic norm of difference between several (n_c) cycles of the network:

$$K_{HD} = \arg \min_{K_{HD}} \left\| \mathbf{A}^{HD}(t) - \mathbf{A}^{HD}(t + n_c) \right\|_2. \quad (5)$$

A modified sigmoid function $\sigma(\cdot)$ was used similar to those presented in [18]:

$$\sigma(x) = \ln \left(1 + \frac{e^{x-b}}{a} \right). \quad (6)$$

The reason of using the sigmoid function in this form is the better accordance of the resulting neural activity than in case when using sigmoid or hyperbolic functions.

2.4 Place cells

Our model simulates function of place cells (PC) [9] in the hippocampus. PC is a neuron which is active whenever the animal is situated in a specific part of the space. Set of positions where the cell is active is referred as a "place field".

We design a recurrent neural network analogous to those in the HD system. The idea to extend HD system to create a two-dimensional spatial map is presented in [18]. For each place cell i , we define a center of the place field as a 2D vector of coordinates \mathbf{X}_i . Coordinates of the place field centers are aligned to a grid. Like in the HD system, we create a synaptic matrix \mathbf{W}^{PC-0} which is constructed analogously in the 2D space.

2.5 Path integrator

PC store animat's supposed position. Our animat is capable to update its supposed position in the environment during locomotion. This provides the path integrator (PI) (fig. 2). The animat doesn't need the sensory (allothetic) inputs to determine its actual location in each moment. Instead, it moves as it would be blind most of the time (see fig. 7). This is because the processing of the sensory inputs is a time-consuming operation. The latter is done only once in a while to correct differences between real and supposed positions

that rise from bad estimates of the heading and speed and integration errors.

We define matrices \mathbf{W}^{PC-N} , \mathbf{W}^{PC-S} , \mathbf{W}^{PC-E} and \mathbf{W}^{PC-W} to move the coded location to the four cardinal points:

$$w_{i,j}^{N,S,E,W} = g(\text{dist}(\mathbf{X}_i, \mathbf{X}_j + \mathbf{B}^{N,S,E,W})) - w_{i,j}^0, i \neq j; \quad (7)$$

$$w_{i,i}^{N,S,E,W} = 0$$

Symbols $\mathbf{B}^{N,S,E,W}$ represent unit vectors in directions of north, south, east and west. These matrices present PI neurons with activity correlated with both heading and actual position. The PI neurons get signals from three sources: HD, PC and their activity is triggered by motor signals. To activate a PI neuron, the animat must move.

In contrast to the HD system, where synaptic weights between HD neurons are modulated by locomotion, PI is a complex subsystem composed by many neurons.

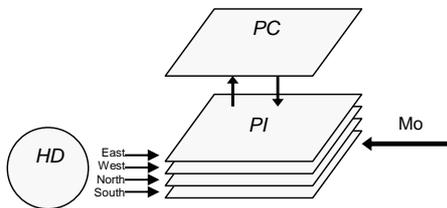


Fig. 2: Schema of the path integrator (PI). The PI consists of 4 groups of neurons to move supposed position to the east, west, north and south. Synapses between HD system and the PI system select one or two of these groups; so that it's possible to shift the supposed position to an arbitrary direction. In each group, only cells corresponding to actual location are activated by PC, and the activity of these cells is triggered by vestibular motor (Mo) signals.

2.6 Place recognition

Most of the time, the animat moves in the environment as it would be blind and it uses PI to obtain its position. From time to time, the animat stops moving and “looks around”. Then, it updates its supposed position in the arena. The SI affect both HD and PC systems to correct an inaccuracy cumulated by PI. But this is possible only when the animat remains in a known environment. On the other hand, when the actual position is supposed to be unknown, the animat should associate actual sensory information with the currently supposed position. Unfortunately, this supposed position contains the inaccuracy cumulated by PI.

Animals solve this problem by iterative returning to known positions [13]. It helps to maintain the inaccuracy cumulated by PI small. It was not our aim to solve these problems so we avoided it by pre-wiring a neural layer from SI downstream to the HD and PC systems, referred as place recognition (PR) system. It conforms to the concept that the animal moves in a well-known environment, as it was in Morris water maze [8].

2.7 Navigational map

Now, we describe, how our animat solves a simple navigational task. Consider two grids of neurons. The

first is the formerly presented PC system. The second is a duplicate of the first one where the navigational map (NM) will be created. Before starting of solving of the task, the neurons in the latter grid code actual animat's location, as the PCs do. During the learning process, after a couple of successful retrievals of the target, the position coded by the NM grid will shift towards the target. Whereas the PC code animat's actual location, our NM cells code location closer to the target and the difference between these locations determines direction of the further move.

We use a similar approach to that presented by [3], based on the long-term potentiation (LTP) and the reinforcement learning. In each step, we compute a “potential” of every cell by equation

$$P_i^{NM}(t) = \max(A_i^{NM}(t), k_{d-NM} \cdot P_i^{NM}(t-1)); \quad (8)$$

$$P_i^{NM}(0) = 0,$$

where k_{d-NM} is the decay factor in the range 0÷1, and $A_i^{NM}(t)$ is an activity of the NM-cells, analogous to the activity of PC and similar to activity of HD in (4).

There is a difference between the PC and the NM cells. Whereas the PC-to-PC synapses (matrix \mathbf{W}^{PC}) remain fixed, the NM-to-NM synapses (analogous matrix \mathbf{W}^{NM}) change by the learning process in the following manner:

$$\Delta w_{i,j}^{NM} = r(t) \cdot A_i^{NM}(t) \cdot A_j^{NM}(t) \cdot \max(-\beta, \text{sgn}(A_i^{NM}(t) - A_j^{NM}(t))) \quad (9)$$

Consider i to be an index of the postsynaptic cell and j index of the presynaptic one. Change of the synapse between two NM cells is proportional to the potentials of the both cells. The synapse reinforces, if potential of the presynaptic cell is bigger then the potential of the postsynaptic one (fig. 3). It happens, if the presynaptic one fires sooner then the postsynaptic one. In a reverse case, the synapse weakens, but this influence is β -times smaller (we choose values of the β parameter in range 0÷1). This parameter is more comprehensively discussed in [3]. The last term in (9) is a function $r(t)$ – the reinforcement signal well known from the reinforcement learning theory. The learning process described above has effect only in the case of the non-zero $r(t)$. This signal is zero most of the time, so the learning process isn't running. The signal is set up to the positive value in the case of reaching the target or a familiar environment (it means a place from where the path to the target is well known). This approach is known as delayed reward.

2.8 Locomotion control

Locomotion control (LC) subsystem is used to read data stored in the NM. Direction of the next animat's step is computed by comparing NM cells activity with PC activity. In the case that NM map is learned enough, both activities significantly differ and right movement

direction can be determined; animat is exploiting its NM. If both activity maps coincide, the map is not suitable for the navigation and the movement direction is not changed so that exploration is performed.

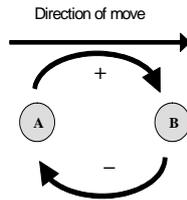


Fig. 3: The LTP when building the navigational map. Consider two NM-cells, A and B, with place fields as shown at the picture. When the animat moves, the cell A fires before B, and the A-to-B synapse reinforces, and the B-to-A synapse reduces.

3 Simulations

We used our animat to solve a simple navigational task in both convex and non-convex environments. The task was similar to Morris water maze [8]. An “invisible” circular target was placed in the arena. When the animat reached the platform, the task was accomplished and the animat received positive reward signal.

The animat evaluated direction of the further move in each step. If the animat’s NM had been learned enough, where to go, the animat followed its supposed direction. Otherwise, it continued in the current direction. When it bumped into the arena wall, direction of the next step was randomly chosen towards the arena.

In some cases, static rectangle-shaped obstacles were also present in the environment. Bumping into an obstacle was similar to bumping into the wall: a random feasible direction was chosen. Hence, the animat also had to learn how to by-pass the obstacles. It made the task more difficult: the shortest trajectory to the target was not simply the direct one, but more complex trajectories had to be chosen.

3.1 Parameters

Consider our arena was a rectangle 100x100 units and one landmark in each corner of it. The animat was capable to detect relative bearing (distance and angle) of all of the landmarks simultaneously. The target was a circle 20 units in diameter. Animat’s speed was normalized to 3 units per step.

Our neural network system consisted of several parts containing in sum nearly 6000 neurons. We modeled SI by creating 36 egocentric-bearing-sensitive and 16 distance-sensitive neurons for each of the 4 landmarks in the corners of the arena. HD system was made by $N_{HD}=100$ neurons and PC formed grid 30x30 neurons ($N_{PC}=900$). The largest part, PI, consisted of $4 \times N_{PC}=3600$ neurons. 1000 neurons were used for PR. Additional neurons were used for LC, activity control of the network etc.

For our modified sigmoid function $\sigma(\cdot)$ defined by equation (6), we chose following parameters: threshold $b=0.3$ and $a=\ln(e^1 - 1)/1 - b=0.7733$ for normalization $\sigma(1)=1$ and $\sigma(0)\approx 0$. Other parameter values mentioned earlier were: $K_{HD}=0.19$, $K_{PC}=0.039$, $\beta=0.7$, $k_{d-NM}=0.7$.

3.2 Individual Subsystems: HD, PC and PI

The HD system was supposed to form an activity packet, which coded animat’s heading. The activity packet had to maintain it in the case of zero sensory input without any drift. We initialized the network activity $A^{HD}(0)$ by uniformly distributed random noise in range 0.0÷0.1. In about 5÷10 iterations of (4) the activity packet appeared and later, no significant changes nor drift were observed during next hundred iterations. In practice, noises in cell firing or synapses might cause the packet to drift [18].

We observed how the HD activity packet moved in the case of discrepancy of coded heading and sensory (allothetic) information. Consider the HD system was coding animat’s heading. Suddenly, new inconsistent allothetic information was introduced by SI into the HD. There’s a discrepancy between supposed and real headings. When the discrepancy was low, the activity packet continuously moved towards the new position. When the discrepancy was too high, the old activity packet disappeared and new one currently appeared. These results are similar to those published in [18].

Analogous results (activity packet appearance and remapping) were observed in PC. Compared to HD, greater effort and more iteration were necessary.

We also measured an accuracy of HD integration capabilities (3). By simulations we observed that the inaccuracy wasn’t more than 1% for the working range $v_{HD}=0.1\div 1$. The results were worse for $v_{HD}>1$. Another solution brings [14]. It contains different groups of neurons to shift the activity packet for different angular velocities. Mechanism of shifting of the activity packet by vestibular signals in brain still remains a question.

Function of the PI can be seen in fig. 7. The PI is able to predict animat’s position, but it cumulates an inaccuracy, mostly caused by bad estimate of heading. Iterative recalibration helps to limit the inaccuracy.

3.3 Navigation through the Environment

The resulting spatial map in the case of convex environment can be seen in fig. 4. The animat was capable to learn to navigate in a major part of the environment in 15÷20 episodes (fig. 5). No significant changes in the map were observable after more than 20 episodes except of arena corners and the most distant places from the target.

The case of non-convex environment (containing an obstacle), can be seen in fig. 6. The number of the episodes necessary to certain navigation depended on the

shape of the non-convex environment. Although the animat was able to solve all exposed situations, problems with bypassing of the obstacles were observed. Despite of the fact the animat once learned sufficiently to bypass the obstacle and not to bump to it, it tried to cut short the trajectory in later episodes, which resulted in bumping to the obstacle again. Note that the obstacle couldn't be detected by any sensor and that the only information stored in the memory was a file of recently passed trajectories coded in the navigational map.

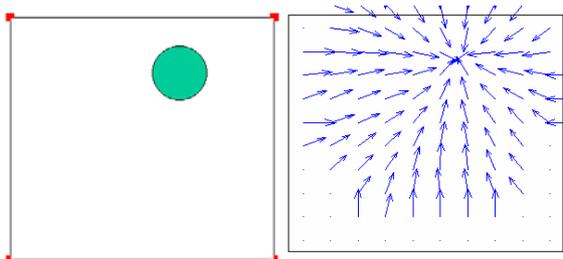


Fig. 4: A convex environment with landmarks in corners and a sample navigational map within this environment. Arrows point towards the target.

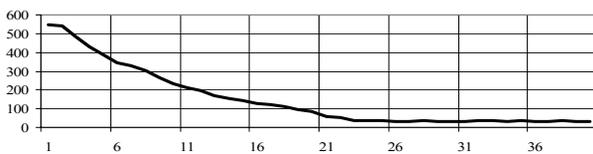


Fig. 5: Learning in a convex environment. Horizontal axis represents an index of episode, vertical axis the average number of steps to reach the target from an arbitrary position.

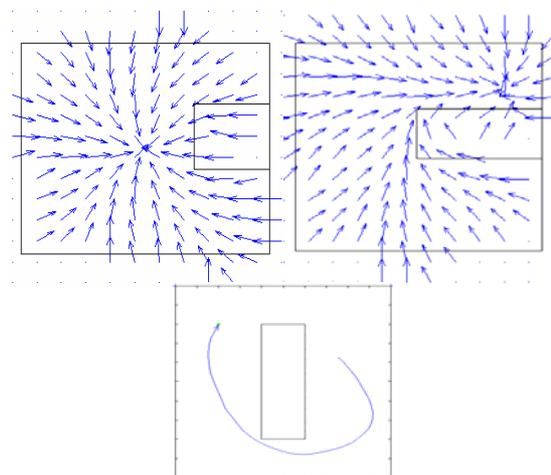


Fig. 6: Sample navigational maps and a sample trajectory produced by the animat in non-convex environments. Arrows point towards the target.

4 Conclusion

4.1 Biological Plausibility

Despite of many simplifications, our model implements building blocks expected to be presented in neural circuits of animals' hippocampal formation. It models such phenomena as place or head direction cells. It also

incorporates the path integrator. Model inputs, sensory and vestibular information, are also modeled by biologically plausible way. Our primary aim was to model outer behavior rather than internal processes of each block. Hence some phenomena are modeled only from the outer view or aren't modeled at all.

The sensory information used to locate the animat in the environment consists of information about distance and direction of the spatial landmarks. It seems to be oversimplified but this approach is frequent in many models. On the other hand, from the biological point of view it is not unreal. Both animals and humans use egocentric-bearing related information for navigation [15]. It is also generally acknowledged that rodents don't process visual information continuously but only once in a while when they look around. Similarly behaves our model.

PC and HD are modeled by recurrent attractor neural networks as in many other works [18], [14]. Some models [6] use spiking model of neuron instead of our simple mathematical model. However, usage of such a neuron models for real-time robot navigation is constrained by performance of computers so that looking for less demanding approaches is also necessary.

Biologically implausible is the regulation mechanism of neural activity in HD and PC systems (4). We should model inhibitory interneurons rather by a set of inhibitory cells. It would be likewise to real brain circuits.

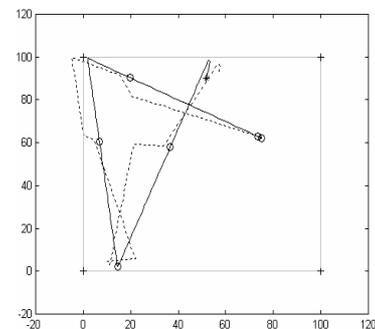


Fig. 7: Function of the PI. Solid line represents animat's location in the arena, dashed one is a trajectory computed by PI cells activity. Circles denote places of recalibration of the animat's position.

4.2 Solving of the Navigational Task

Our animat is capable to solve a simple navigational task in both convex and non-convex environment, although the resulting trajectories are far from optimality (fig. 6). In a convex environment, a couple of episodes is enough to learn to solve the navigational task. This is comparable with experiments on rodents in Morris water maze [8]. A comparison of our animat's behavior in a non-convex arena is problematic because no experimental data with a hidden platform and an invisible obstacle are available.

Somebody could argue that such a task could be solved (excepting path integration) by a small number of

neurons. In early eighties several models were published, e.g. [2], containing as less as 4 neurons solving similar task and using reinforcement learning. The main differences between our (and other up-to-date ones) and Barto's models are (i) ability to learn from delayed reward instead of continual information about distance to the target, (ii) usage of path-integration capabilities to make navigation more robust and (iii) capability to avoid simple obstacles in the environment. On the other hand, nearly 6000 neurons of our model is about hundred times less than rodents' hippocampus is suggested to contain.

Let's compare our model with the others mentioned. [14] is a large and sophisticated theoretical work containing also proposals for navigation in multiple environments. It doesn't consider interactions between the environment and the model nor solving navigation tasks. The model in [1] was successfully tested on a real robot even in multi-target tasks, but it doesn't implement HD or PI. The learning algorithm totally differs. The model in [5] is able to avoid obstacles and tries to clarify role of the θ precession, but also doesn't involve HD or PI. The model in [7] extends the classical model of PC and HD by developing "pose cells" that code both position and heading. Although there is no biological justification for it, the system was successfully tested in an indoor environment with landmarks.

Our model was designed to solve a navigational task in a single environment with a single target like most experiments do. However, for use in a robotics domain, we need our model to learn route to multiple targets and to distinguish multiple environments. These will be subjects of our future work.

Acknowledgments

This project is supported by Czech Ministry of Education, Youth and Sports FRVS grant 758/2005.

Thanks to the researchers of Institute of Physiology Academy of Sciences for their useful suggestions

References:

- [1] A. Arleo, W. Gerstner: Spatial cognition and neuro-mimetic navigation: A model of Hippocampal Place Cell Activity. *Biological Cybernetics*, Vol. 83, No. 3, 2000, pp. 287-99
- [2] A. G. Barto, R.S. Sutton: Landmark learning: An illustration of associative search. *Biological Cybernetics*, Vol. 42, No. 1, 1981, pp. 1-8
- [3] K. I. Blum, L. F. Abbott: A model of spatial map formation in the hippocampus of the rat. *Neural Computing*, Vol. 8, No. 1, 1996, pp. 85-93
- [4] J. Bures, A. A. Fenton, Y. Kaminsky, L. Zinyuk: Place cells and place navigation. *Proc. Natl. Acad. Sci. USA*, 1997, Vol. 94, pp. 343-350
- [5] N. Burgess, M. Recce, J. O'Keefe: A Model of Hippocampal Function. *Neural Networks*, Vol. 7, 1994, pp. 1065-1081
- [6] T. Degris, L. Lachèze, C. Boucheny, A. Arleo: A Spiking Neuron Model of Head-Direction Cells for Robot Orientation. In *Proceedings of the Eighth Int. Conf. on the Simulation of Adaptive Behavior, from Animals to Animats*, MIT Press, 2004, pp. 255-263
- [7] M. J. Milford, G. Wyeth, D. Prasser: RatSLAM: A Hippocampal Model for Simultaneous Localization and Mapping. In *proceedings of International Conference on Robotics and Automation*, New Orleans, United States, 2004
- [8] R. G. Morris, pp. Garrud, J. N. Rawlins, J. O'Keefe: Place navigation impaired in rats with hippocampal lesions. *Nature* No. 297, 1982, pp. 681-683
- [9] R. U. Muller, J. L. Kubie, E. M. Bostock, J. S. Taube, G. J. Quirk: Spatial firing correlates of neurons in the hippocampal formation of freely moving rats. In J. Paillard: *Brain and Space*. Chapter 17, pp. 296-333, Oxford University Press, New York, 1991
- [10] J. O'Keefe, J. Dostrovsky: The hippocampus as a spatial map. Preliminary evidence from unit activity in the freely-moving rat. *Brain Research*, Vol. 34, 1971, pp. 171-5
- [11] J. O'Keefe, L. Nadel: *The Hippocampus as a Cognitive Map*. Clarendon Press, Oxford, 1978
- [12] C. Pavlides, Y. J. Greenstein, M. Grudman, J. Winson: Long-term potentiation in the dentate gyrus is induced preferentially on the positive phase of θ rhythm. *Brain Research*, 1988 Vol. 439, pp. 383-387
- [13] A. D. Redish: *Beyond the cognitive map*. School of Computer Science, Carnegie Mellon University, Pittsburgh, 1997
- [14] A. Samsonovich, B. L. McNaughton: Path integration and cognitive mapping in a continuous attractor neural network model. *Journal of Neuroscience*, Vol. 17, No. 15, 1997 pp. 5900-5920
- [15] H. Schone: *Spatial Orientation: The Spatial Control of Behavior in Animals and Man*. Princeton University Press, 1984
- [16] A. Speakman, J. O'Keefe: Hippocampal complex spike cells do not change their place fields if the goal is moved within a cue-controlled environment. *European Journal of Neuroscience*, Vol. 2, No. 6, 1990, pp.544-555
- [17] J. S. Taube, R. U. Muller, J. B. Ranck Jr.: Head-direction cells recorded from the postsubiculum in freely moving rats. I. Description and quantitative analysis. *Journal of Neuroscience*, Vol. 10, No. 2, 1990, pp. 420-435
- [18] K. Zhang: Representation of spatial orientation by the intrinsic dynamics of the head direction cell ensemble: A theory. *Journal of Neuroscience*, Vol. 16, No. 6, 1996, pp. 2112-2126