Navigating with an animal brain: a neural network for landmark identification and navigation.

Philippe Gaussier * , Stéphane Zrehen **
* ENSEA ETIS, 6 Av du Ponceau, 95014 Cergy Pontoise Cedex, France
** Laboratoire de Microinformatique, EPFL-DI, CH-1015 Lausanne
Tel : ++ ++ 1 30 73 66 13
EMmail : gaussier@ensea.fr, zrehen@lamisun.epfl.ch

1 Introduction

Navigating in an unknown environment is a task commonly accomplished by most animals. Nevertheless, it is not justified to infer that this capacity needs complex reasoning involving abstract geometrical computations. Indeed, it is hard to imagine that ants or wasps have such knowledge, while they are clearly able to locate a source of food, and then go back and forth between that source and their home. In this paper, our aim is to show that such behavior, including switching between goals, can be simulated by simple artificial Neural Networks (NN) where no complex computation is performed. We will present a real development and simulations about a Khepera™ robot (fig. 1) and a simulated system named Prometheus.

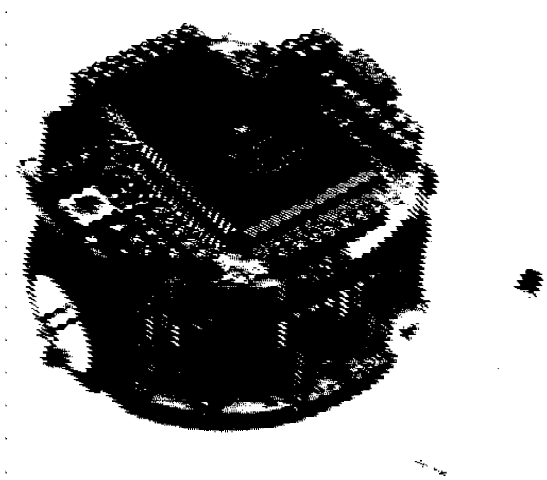

Figure 1: The Khepera™ robot developed at the LAMI [Mon93].

We use a novel neural architecture named PerAc (Perception-Action) which is a systematic way to decompose the control of an autonomous robot in perception and action flows [Gau94b]. We show that action simplifies the interpretation of perception: each action is a choice and conditions entirely the future of the robot. That way, acting in the world is necessary to the categorization and hence the interpretation of perceived signal, i.e., to the emergence of an elementary "cognition". The greatest advantage of this type of approach is that it makes cognition sequential, thereby avoiding the possible large duplications and relaxation mechanisms needed by massively parallel systems.
We also focus on the interest to perform an autonomous on line learning of the relevant places to the robot in its environment. Furthermore, we compel ourselves not to touch modify the internal structures of the artificial robot "brain" by hand, while it operates. Thus we have to:

- design a self modifiable connection diagram.
- pay attention to the self adaptability of each simple block to data variations.
- allow the robot to use the signals correlations which are really relevant to it.
- introduce a limbic system to control the robot's learning, motivation and behaviors.

We emphasize the interest of a constructivist approach [Mat87], [Ste91] as implemented for instance by the subsumption architecture [Bro86]. A special stress is put on introducing goal resolution in our biologically plausible model of vision and navigation system.
We propose a design for an extremely simple limbic system (the one mainly involved with emotions in our own brains), in order to deduce the overall structure of the neuronal basic element. This leads us to use the concept of simulated cortical column first proposed by [Bur89].
In a first part, we briefly sum up the characteristic of the PerAc architecture and we show how it can be

used to extract localization information from a visual scene. Next we show how a robot can learn to return from any starting point to a previously discovered and learned position without any a priori symbolic representation. At last, we simulate a complete behavior consisting in avoiding dangerous zones to go to "eat" and next to return at home when the robot is "tired."

2 The PerAc building architecture

We have already realized robot 'brains" with simple conditioning that allows them to learn sensory situations and at the same time to learn what movement must be performed. For instance, when the robot collides in a wall a pain signal is emitted and a reinforcement learning rule is used to increase the synaptic weights in order to avoid obstacles on following similar situations [Gau94a]. The same mechanism is involved to recognize objects. The robot's eye is able to move its eye from one point to the other in a perceived scene [Gau92]. It learns an object as a sequence of local recognitions associated to particular ocular saccades. All this system is simulated with unsupervised neural networks. The output of the visual system, that is, the local recognitions and the angular movement to go from one focus point to the other are used as inputs to another part involved in targets retrieval. The local recognition is associated to the identification of landmarks in the visual environment and the ocular movements provides information about the angle between two landmarks. A simple neuron called "place cell" can then learn a particular location [Zip85] and react according to the proximity of the robot to this stored location.
In our system, all the neural groups involved are modelled by self organized topological maps [Gau94a]. This implies that our robot is able to store new information near similar ones previously learned. Then, a lateral diffusion mechanism allows to react to the new learned shape in the same way as for the previous ones. Whenever the reaction is wrong, the robot can learn to refine is classification according to the action that must be performed.
Prometheus' “ brain ” architecture is summarized on fig. 2. The same neural architecture is used to recognize an object or a landmark and to control the robot movements. The PerAc blocks of which it is made appear to be a kind of basic building block and a systematic tool to combine motor and perceptive information. Perac architecture relies on the postulate that the recognition of any cue can be simplified if the system can act on it. This justifies to cut any perceived cue into two parts: a) a motor part which is the result of a hardwired conventional processing and b) a cognitive one which proposes to learn/record important situations and to allow a quicker adaptation of the system's response.

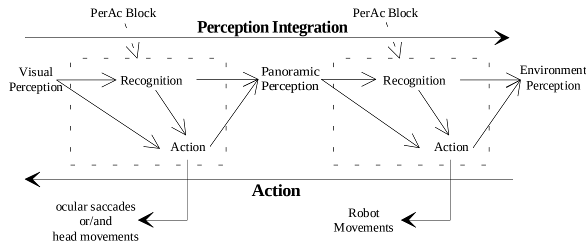

Figure 2: The complete neural network for target retrieval modelled by PerAc blocks

Our model also offers an alternative to the classical scheme of hierachical classification because we integrate not only static perceptive recognition information but also motor information provided by the input cue or/and the local recognition.
Navigation problems are a good example for illustrating the problems to manage goal achievement and switching in a completely autonomous system controled by a single neural network without any programmed trick to allow the good choice at the right time. The solution we propose can be understood on fig. 3.

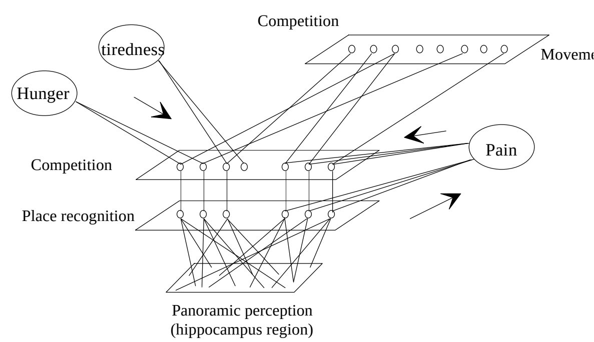

Figure 3: Intuitive structure of a N.N. that allows the switch between different motivations.

The neural network is composed of a recognition part with a Winner Takes All (WTA) competition mechanism that allow to recognize a situation and to inforce the activation of a particular movement according to what has been learnt. The different motivation nuclei allow to favour a particular sub group of recognition neurons. In the next part, we will describe how a such basic network can explain navigation behaviors. Next, we will return to the problem of behavior switching, and therefore we will study how to link properly the different motivation nuclei with the simple PerAc N.N. structure.

3 Target retrieval using landmarks

At the beginning, we suppose Prometheus moves randomly. When it finds "food", it moves around it and learns that from several particular locations it can go to the target by performing a movement in a given direction.

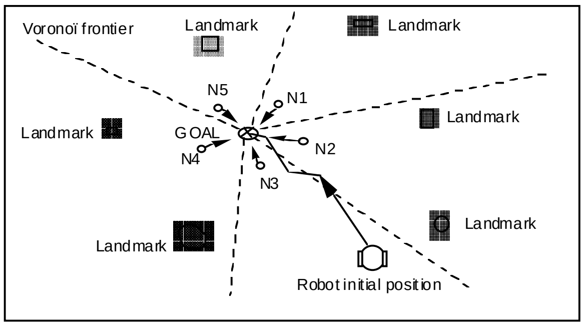

Figure 4: Local exploration around a target.
The target is represented by a circle at the intersection of the dotted line representing the associated recognition domain of Ni neurons. The robot records at certain points (represented by small circles Ni) their relative position to the landmarks and the direction to the target.

Later, when Prometheus wants to find "food", it considers the information of the place cells associated to the food and goes in the direction associated to the most activated place fields (competitive mechanism). Thus at each time, the distance to the target is reduced (fig. 4) and Prometheus is bounded to return to the learned position of the food. The complete "brain" is depicted on fig.5. The local visual recognition (LR) and the information about the eye movements (EM) can be joined to provide information about " what " the landmarks are and " where " they are from each other. Simple product or logical AND neurons can be used to merge those different information type in a map of neurons that reacts only if a particular landmark is recognized at a particular place: GVI group (Global Visual Input) in fig. 5. Matching between a proposed visual scene and a learned scene is performed with a topology preserving map.

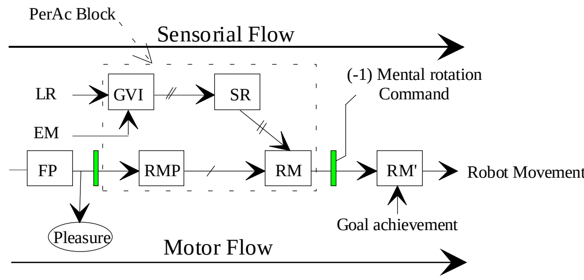

Figure 5: The navigation neural network.

When a movement direction is selected (RM': Robot Movement), the robot makes one step of given length in that direction. The input to this network are the north direction, and the food and landmarks positions in the robot's visual space. We assume that a compass is available. It could be replaced by a vestibular system or a gyroscopic mechanism that would produce low precision information about the body orientation (a local landmark could also be used but it would reduce the generalization capabilities of the robot to very distant situations).
This system allows target retrieval when the place cells have been learned. We have proposed a neurally-coded reflex mechanism [Gau94b] that allows to visit several evenly disposed places around the target, which includes a pleasure-linked regulation for learning control.

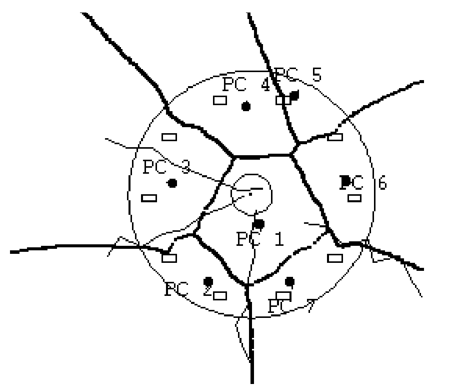

Figure 6: Different trajectories.
The Place-cells (PC) are indexed by their order during exploration. The Voronoi tessellation is represented by the thick lines, the landmarks by the rectangles and the target by the inner circle. The large circle represents the limit beyond which the target is not perceived. Thin lines represent trajectories from various starting-points.

We have succesfully implemented and tested the neural network described above on a Khepera robot. Due to the tremendous computing time required, we simulate the visual part that has been tested elsewhere [Gau92]. The robot succeeds in learning the food position, and later, it always takes the right direction, whatever its starting point (fig. 6). More realistic trajectories can be obtained if the movement is performed according to a probabilistic vote rather than a determinist WTA mechanism.

4 Avoidance of forbidden areas

The previous mechanism allows Prometheus to go back to a learned position in a somewhat straight line but it does not take into account any zone the robot must not go through. Moreover, forbidden areas not necessarily have an intrinsic reality for the robot. Hence, we need to introduce *ad hoc* mechanisms that allow to generate for instance a pain signal when the robot enters a forbidden area. Such a zone is then perceived as dangerous and the robot uses the same mechanism as for obstacle avoidance to learn the direction to avoid pain. As a result, learning a forbidden zone is equivalent to learning an interesting place. The robot learns different meaningful places where the pain signal is high and also the association to the proper movement in order to avoid pain (see fig. 7).

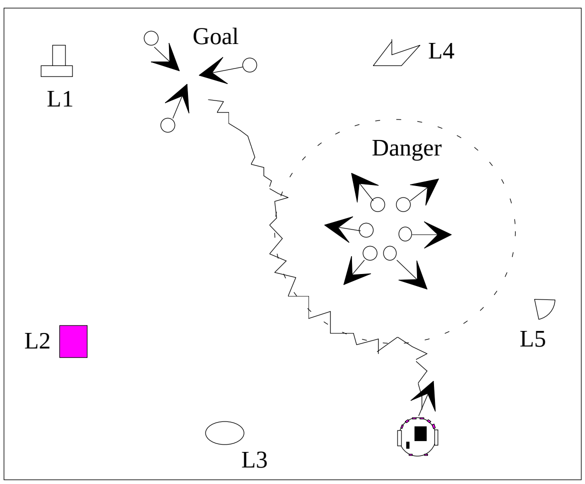

Figure 7: An example of situation with a dangerous area that must be avoided to reach the goal.

With such a system, a problem arises when the robot is closer to the dangerous area than to the food area. Indeed, the neurons associated to the recognition of the dangerous area have a higher activity than those associated to the goal. Then, the most activated of them must win and the robot will perform a movement to avoid the forbidden zone, thus moving in a direction opposite to its goal, which is thus never reached. That problem can be solved if the influence of the neurons associated to "non-goal" actions can be inhibited or switched off when the robot is far enough from the dangerous area. This means that when the recognition of a dangerous position is not high enough, recognition of dangerous areas has no effect on the robot behavior. But how to switch of the avoidance behavior? Moreover, according to the simple retrieval system when the robot has found its target once, it will remain forever in its proximity. So the general question is how the robot itself can be able to switch off its unexpectated behavior.

A first solution to inhibit a particular recognition could have consisted in tuning the neurons' selectivity but it is impossible because this parameter is already used to automatically control the unsupervised learning of the neurons [Gau94a]. Moreover, it seems difficult to change only the selectivity of one particular neuron that interests us and not of the others.

Actually, the problem is that Prometheus should recognize a situation that is not the best fit but that agrees its goal. The solution, we propose is inspired by Grossberg's studies about contour closing in preattentive vision [Gro87] and by Burnod's model of the cortical column and cortical map [Bur89]. Grossberg proposes a structure composed of two layers of neurons with feed-forward and feed-back links. The former is a competitive network with neurons associated to the recognition of a particular situation (i.e: a piece of straigth line) whereas the second level tries to propose positions for contour completion by reacting on the first neuron layer. If the total sum of the input activities is high enough, the neuron on the first layer is activated and a piece of edge is "recognized" at that position even if the initial recognition was not high enough. This N.N. can so be seen as a way to achieve a preattentive goal of contour completion. On a other level, Burnod's formalization of the cortical column distinguishes a thalamic level associated to action choices and a cortical level in which rewarded goals and subgoals can propagate freely according to most frequent transititions between different situations, as in a relaxation or a Markov process.

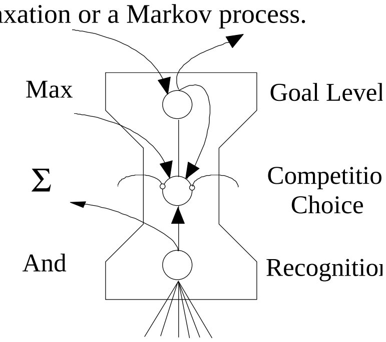

Figure 8: representation of a cortical column

To sum up, our neural building block must have three layers (fig. 8). A first level must be associated to the recognition of the current situation according to their synaptic weights. A second level consists in a competition mechanism that must choose the recognized situation according to the first level information and to those coming from goal. And the third one can be used to propagate the goals and subgoals. Fig. 9 represents such an architecture, which allows to avoid one forbidden area. In addition to the cortical column structure, the limbic system is represented through different neural and chemical nuclei. Those nuclei are linked to the cortical maps in the same way as action units are associated to recognition units in the simpler N.N. When pain is active, the links betwen the neurons representing dangerous situations and the nucleus associated to pain are increased, thus allowing the robot to feel pain earlier when it sees once more a given dangerous situation. We also suppose that when the pain nucleus is inactive, it induces a negative activity to its associated neurons on the goal level of the recognition cortical map. The same thing goes for the hunger nucleus.

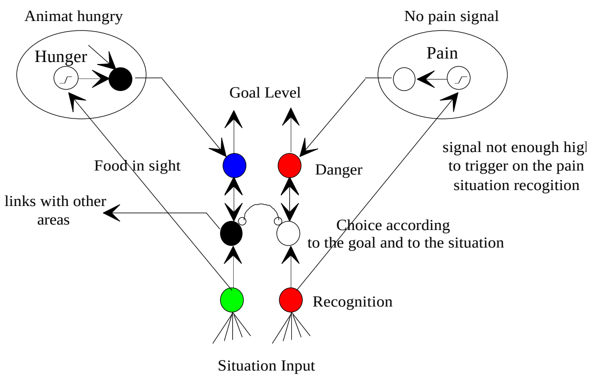

Figure 9: How Prometheus inhibits an avoidance behavior when it is hungry enough.

Next, we have to suppose that the different nuclei take their inputs from the analog value of the recognition and not from the competition. Indeed, they need to have an analog value that really represents the proximity to the learned situation and that can allow to trigger the avoidance behavior if the situation is too similar to the painful situation learned. For instance, on fig. 7, we suppose the robot is closer to the forbidden zone than to the food. Thus, the most activated neuron on the recognition level is one associated to the forbidden area. But its activity may not be high enough to switch on the pain nucleus, whereas the hunger nucleus is activated because the robot "needs" food. Then, the goal level associated to the recognition of the food places can allow them to win and to induce the movement in the good direction, ie: to go to the food. If both nuclei are inactive then the associated recognition neurons cannot be activated and no information is provided to the motor group. Hence, the robot moves randomly.

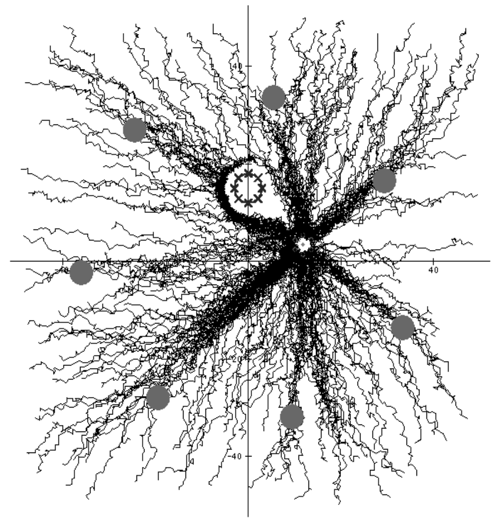

Figure 10: Simulation of a forbidden area avoidance and goal reaching.

Fig. 10 represents a simulation in which the robot is hungry. It goes in the direction to the food since it is too near to the forbidden area. Then, the avoidance of pain is activated and the robot goes out to the influence domain of pain nucleus and it tries again to reach the food. The probabilist WTA mechanism prevents it from being blocked and allows it to move around the forbidden area.

5 Choosing between different goals

We have shown that a limbic system is needed to control the switching of behaviors between a goal and a prescriptive constraint such as the avoidance of a dangerous area. Now, we are interested in how to switch from one behavior to another. For instance, after having found food, the robot may want to go home. According to the previous diagrams, the robot should stay forever near the food because the recognition of the place associated to the food is very high. Having already eaten should reduce the intensity of the will to move towards food. The will to see home should favour the recognition of the associated places to allow the good movements.

The mechanism presented in the previous part can be extended to choose between different simple goals that must be associated to different internal motivations. According to the activity of each motivation nucleus, the competition mechanism on the recognition cortical map should allow the robot to choose the most interesting behavior that can be performed according to the situation and to its motivations. The neuron activation rule used is the difference betwen the maximum of the excitatory and inhibitory input links. This activity (S_i) is put to 0 when it is negative. Hence, a constant input allows the unreached neurons to win in front of inhibited neurons.

$$S_i = \left[I + Max \left(positive\_activation \right) Min \left(negative\_activation \right) \right.$$

$$if\ x > 0\ \ [x]^+ = x\ else\ [x]^+ = 0$$

The Max operator is a way to make the neuronal activity independant of the number of input links and to give rise to stability of the neuronal activity through the cumulative effect of feedback positive links. If the weight is lower than 1 there is no divergence problem.

The goal level can also be used to add capabilities of plan generation. When a goal is proposed, if it cannot be satisfied possible subgoals are proposed until a subgoal is really satisfied. Then the actions can be performed according to their goal pathway.

The mechanism can be extended to topological maps [Gau94a] if the inhibition is applied to the whole activity bubble in order to inhibit all the recognition neurons associated to the same movement.

Motor groups use the same law but the simplification comes from the absence of needed recognition. In fact it can be added if actions can only be performed in particular situations or if the choice of the action depends on the position of the motor system; for instance, a mechanical arm with several freedom degrees and forbidden angular zones...

6 Conclusion

We have shown that a simple neural system can be used to control robot navigation and goal management.

Research on these mechanisms should lead to define an explicit parallel langage to “ program ” animal-like robots with adaptation and autonomy capabilities.

The neural vision sytem associated to the robot has not been used in the experiments presented above for reasons of computation time. We are now working on a multi processor architecture that may allow to dispatch simply the explicit parallel program that the N.N. represents. Other work focuses on learning with delays between the conditional stimulus and the unconditional one or the reinforcement signal. At another level algorithms able to learn a succession of perception action according to a latter goal have been successfully tested. Future work will consist in assembling all these part to (really) realize a (animal) robot able to navigate with a real autonomy in an outdoor environment.

7 References

[Bro86] Brooks R.A., A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation* RA-2, p 14-23, 1986.

[Bur89] Burnod Y. , An adaptive neural network : The cerebral cortex, *Masson*, Paris, 1989.

[Gau92] Gaussier P. , Cocquerez, J.-P., Neural Networks For Complex Scene Recognition: Simulation Of A Visual System With Several Cortical Areas, *in Proceedings of IJCNN*, Baltimore, Vol. 3, p 233-259, 1992.

[Gau94a] Gaussier P. & Zrehen S., A Topological Neural Map for On-line Learning: Emergence of Obstacle Avoidance in a Mobile Robot, *SAB 94*, Brighton, not yet known, 1994

[Gau94b] Gaussier P. Zrehen S., A Constructivist Approach for Autonomous Agents, *Wiley & Sons*, Ed N. Thalman, 1994

[Gro87] Grossberg S., Mingolla E., Neural Dynamics of Surface Perception: Boundary Webs,Illuminants, and Shape-from-Shading, *CVGIP No 37*, p 116-165, 1987.

[Mat87] Maturana & Varela, *The tree of knowledge*, Shambhala ed., 1987.

[Mon93] Mondada F., Franzi, Ed., Ienne P., Mobile Robot Miniaturisation: A Tool For Investigation In Control Algorithms, Proceedings of the Third International Symposium on Experimental Robotics, Kyoto, Oct. 28-30, 1993.

[Ste91] Stewart, J., Life = Cognition: The epistemological and Ontological significance of Artificial Life, *proccedings of SAB 91 Paris*, Bourgine P. & Varela F. eds, MIT Press, pp 475-483, 1991.

[Zip85] Zipser D., A Computational Model of Hippocampal Place Fields, *Behavioral Neuroscience* 99, 5, p 1006-1018, 1985.