

Self-organization of reconfigurable processing elements during mobile robots missions

Laurent Rodriguez, Jérôme Fellus, Benoît Miramond
 ETIS Lab UMR 8051 CNRS / ENSEA /UCP
 6 Avenue du Ponceau
 Cergy-Pontoise
 Email: {firstName.lastName}@ensea.fr

Abstract—This paper describes a bio-inspired architectural approach to design highly adaptive systems in the context of mobile robotics. The concerned robots evolve in an indoor unknown environment and then exhibit several behaviours such as landscape learning, obstacle avoidance, path planning, sensorimotor control. We aim at designing the intelligent embedded controller of those robots. The controller will be able to self-organize its elements in order to adapt its architecture to the robot behaviour. We focus in this paper on the description of the neural network which was developed to self-organize the system. We present the simulation results of our method.

I. INTRODUCTION

The aim of this work is to design an intelligent embedded controller that will be able to self-organize its elements in order to adapt its architecture to the robot behavior.

Inspired from the feature integration theory [1], the robot will use three saliency maps. These three maps provide the robot with a sensorimotor cognitive capability in order to react and to adapt its behavior to the environment.

The controller will be able to adapt the size of the different maps according to the states of its actuators and to the saliency of the information from the external environment. As a result of the Kohonen auto-organizing map model [2], the three saliency maps are constantly competing for the resources of the controller.

In section II we describe the hardware reconfigurable architecture of the controller. After that, in section III we present the self-organizing properties of this system and its behaviour in the context of robotic applications. Then in section IV we present the simulation's results that characterize the behavior of the controller as conclusion.

II. CONTROLLER ARCHITECTURE

The controller is implemented as a single system-on-chip embedded into the robot body. During its life time, the robot exhibits several behaviours that are considered as concurrent processing tasks for the controller.

The main goal is then to adapt the number of processing elements of a regular architecture to the current urgency of a task, where urgency is a function of the input-data activity (saliency) as described in section III.

The adaptation mechanism is entirely distributed, thus the tasks placement onto the architecture resources is computed

on-line by the system itself and does not require any external decision making (local or deported operating system).

The idea is to implement cells as reusable elements which behave as competing neurons in their basic configuration, but can be used by a task as part of its processing. So, each task is implemented as a set of cells. The set of unused cells can be colonized as additional resources to implement bigger tasks. The growing cell sets are thus restricted by the competition between their border cells (see Fig. 1). Since competition is computed from the activity of neighbour neurons (and from input saliency), the geometry of a task in the architecture is not limited to a 2D-rectangle as for standard reconfiguration, but by the number of competing tasks.

As depicted in Fig. 1, the architecture of a cell is thus composed of: direct wiring with its neighbours, a local programmable data-path, a local memory storing program and data, and a configuration port controlled by the local neuron. Four types of information are exchanged in the architecture: the input sensors data on which are computed the different tasks, the results of each processing broadcasted to the neighbours, the program of a replicating cell, and the activities of neurons. The role of these information flows in the self-organizing process is described in the next section.

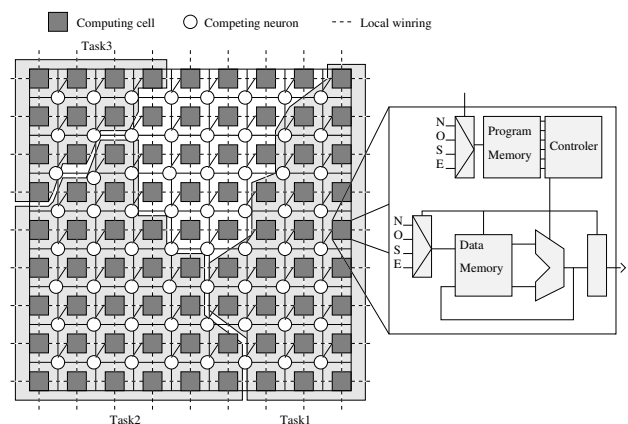


Fig. 1. Architecture of the self-organizing maps.

III. SELF-ORGANIZING MAPS

The concept of self-organization is intrinsically bound to the definition of the Kohonen's maps (p 119 [2]).

This is particularly interesting for an artificial vision application. The self-organizing map (SOM) is able to order its elements according to the entry set. This behavior is used to allocate special areas of a computing substrata to the most suitable tasks. As a result of this behavior, we get a data driven allocation process that can be applied to any application fitting this model.

Intrinsically, it is the case of this robotic application where the vision processing is one of the main cognitive action of the robot. For instance, the artificial vision system of our robot stands on a spatiotemporal visual saliency model [3]. In this model [4], the information contained in the input frames are divided in two types: static and dynamic. The first step of the vision process consists in the extraction of the quantity of each of this type of information in order to balance the allocation through the SOM (c.f. Fig. 2). This can be done thanks to a retina-like neural network where a first ON-OFF layer computes the magnitudes of the spatial gradient and a second layer computes the temporal gradient. The robot is also aware of its actuators states. It takes this information into account through another neural network which feeds the third entry of the system shown on Fig. 2.

As the developed SOM converges to an organization that fits the needs of the application, three main areas (in this case study) emerge from the SOM. Each of them are specialized in relation to the different inputs of the system. For instance, the first and the second areas are allocated to compute the static and the dynamic saliency map. The result of the vision maps is to extract the interest points in the image in relation with the shapes and motions in the scene. Finally, a third area is used as a sensorimotor map. This area will merge the informations from the first two areas and from the actuators of the robot in order to learn and to adapt its behavior relatively to its actions and its environment.

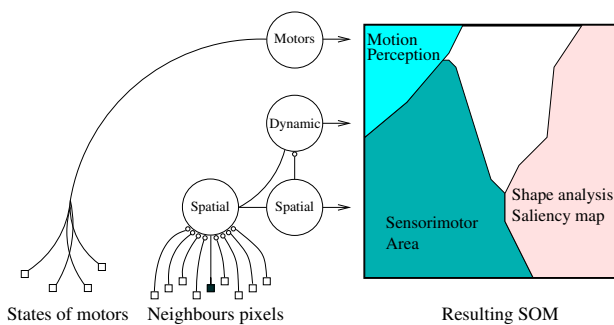


Fig. 2. Binding from the entries of the neural network to the self-organizing map

The learning process of our SOM can be computed as a cost function minimization problem of the average similarity distance between the input vector and the weight vector. The Manhattan distance in the network topology space between the

elected neuron and the learning neuron is used as a criteria of weightiness. A classical "Mexican hat" distribution is then used as a degree of lateral modulation to activate the neurons in a fixed neighborhood and inhibit farther neurons. This choices improve the reactivity of the system and ensure the topological coherency of the network during the adaptation step as illustrated in the following simulations.

IV. RESULTS AND PERSPECTIVES

We developed a simulator of the architecture presented in the previous sections. This simulator can work either with real-time sensors informations from the robot or with recorded data. We presented to the controller architecture the data corresponding to a 3-stages robot indoor mission: A) stationary initialization, B) moving for room exploration and C) stationary landscape learning.

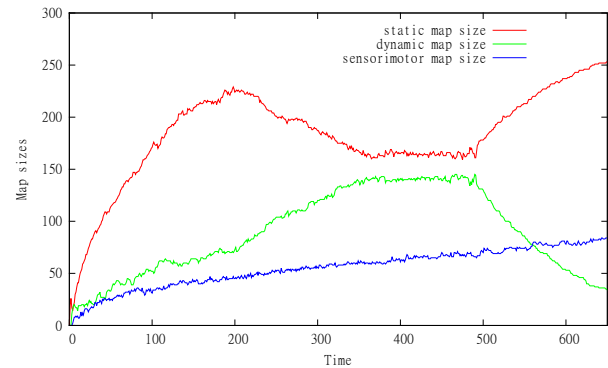


Fig. 3. Binding from the entries of the neural network to the self-organizing map

As we can see in Fig. 3, the maps compete for the resources (cells) available into the controller according to their inputs. The three stages of the robot mission are clearly visible. From the first to the 200th frame (stage A), the robot remains static and a few objects are moving in front of it. From frames 200 to 490, the robot is moving (stage B). The motion estimation map takes space to the static map. This ensure enough computing power to the motion estimation tasks. Finally (stage C) the robot remains stationary moreover there is no motion in front of it. Therefore the stationary map is gaining space from the dynamic one.

The profiling of the application behavior resulting from simulation is very promising for the next stages of design.

REFERENCES

- [1] A. Treisman, "A feature-integration theory of attention," *Cognitive Psychology*, vol. 12, no. 1, pp. 97–136, Jan. 1980. [Online]. Available: [http://dx.doi.org/10.1016/0010-0285\(80\)90005-5](http://dx.doi.org/10.1016/0010-0285(80)90005-5)
- [2] T. Kohonen, *Self-Organization and Associative Memory*. Springer-Verlag, 1989.
- [3] F. Verdier, B. Miramond, M. Maillard, E. Huck, and T. Lefebvre, "Using High-Level RTOS Models for HW/SW Embedded Architecture Exploration : Case Study on Mobile Robotic Vision," *Eurasip Journal on Embedded Systems*, vol. 2008, p. 349465, 2008. [Online]. Available: <http://hal.archives-ouvertes.fr/hal-00524580/en/>
- [4] A. Rahman, D. Houzet, D. Pellerin, and L. Agud, "GPU implementation of motion estimation for visual saliency." [Online]. Available: <http://hal.archives-ouvertes.fr/hal-00564391/en/>